Technical Report SL-95-8
March 1995

US Army Corps of Engineers
Waterways Experiment Station

Evaluation of the Sensitivity and Limits of the Passive Acoustic Ranging Algorithm for Single and Multiple Helicopters

by Benny L. Carnes, WES

John C. Morgan, Illinois Institute of Technology Research Institute

Prepared for Laboratory Discretionary Research Program
U.S. Army Engineer Waterways Experiment Station

Technical Report SL-95-8
March 1995

Evaluation of the Sensitivity and Limits of the Passive Acoustic Ranging Algorithm for Single and Multiple Helicopters

by Benny L. Carnes

U.S. Army Corps of Engineers
Waterways Experiment Station
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

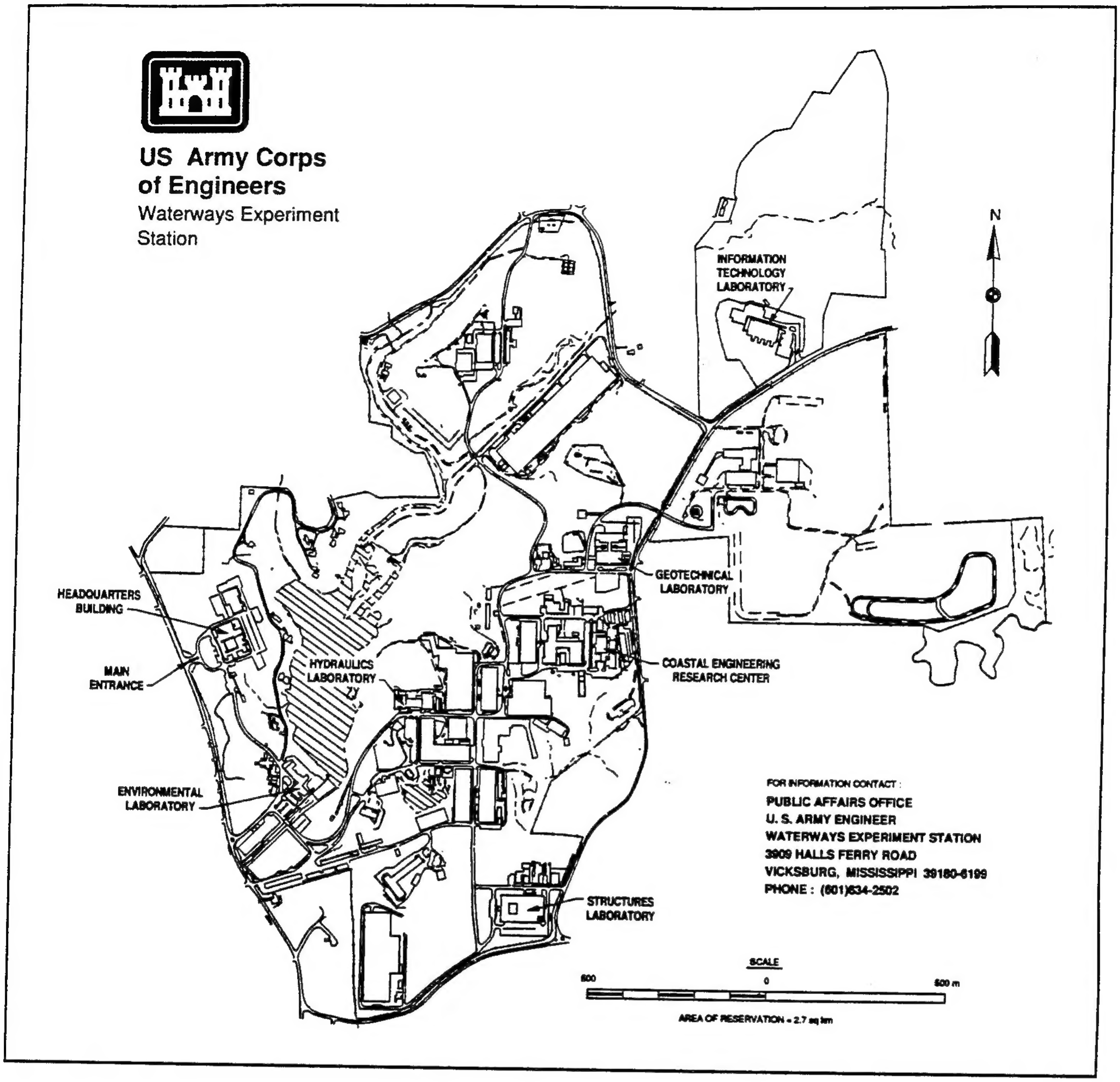
John C. Morgan

Illinois Institute of Technology Research Institute
4140 Linden Avenue, Suite 201
Dayton, OH 45432

Final report

Approved for public release; distribution is unlimited

Prepared for Laboratory Discretionary Research Program
U.S. Army Engineer Waterways Experiment Station
3909 Halls Ferry Road, Vicksburg, MS 39180-6199



Waterways Experiment Station Cataloging-in-Publication Data

Carnes, Benny L.

Evaluation of the sensitivity and limits of the passive acoustic ranging algorithm for single and multiple helicopters / by Benny L. Carnes, John C. Morgan ; prepared for Laboratory Discretionary Research Program, U.S. Army Engineer Waterways Experiment Station.

53 p. : ill. ; 28 cm. -- (Technical report ; SL-95-8)

Includes bibliographic references.

1. Signal processing -- Evaluation -- Testing. 2. Helicopters. 3. Doppler effect -- Military aspects. 4. Military reconnaissance. I. Morgan, John C. II. United States. Army. Corps of Engineers. III. U.S. Army Engineer Waterways Experiment Station. IV. Structures Laboratory (U.S.) V. Laboratory Discretionary Research Program. VI. Title. VII. Series: Technical report (U.S. Army Engineer Waterways Experiment Station) ; SL-95-8.

TA7 W34 no.SL-95-8

Contents

List of Figures

List of Tables

Preface

This study was conducted by the U.S. Army Engineer Waterways Experiment Station (WES), Vicksburg, MS, during fiscal year 1993. It was funded by WES under the Laboratory Discretionary Research Program.

The work was performed under the general supervision of Mr. Bryant Mather, Director, Structures Laboratory (SL), WES, and Dr. Jimmy P. Balsara, Chief, Geomechanics and Explosion Effects Division (GEED), SL, and under the direct supervision of Dr. Benny L. Carnes, Acoustic/Seismic Research Team Leader and Project Coordinator, SL.

This report was prepared by Dr. Carnes and Mr. John C. Morgan, Illinois Institute of Technology Research Institute, Dayton, OH. Field data acquisition was conducted by Messrs. Nathan Larsen, GEED, and Leo Koestler, WES Instrumentation Services Division. Data analysis and preparation were performed by Mr. Morgan and Mr. Travis Harrell, who was assigned to WES from the Computer Sciences Corporation. Preliminary work on the project was performed by Mr. Robert E. Olson, SL. A significant contribution was made in the theoretical physics by Mr. John S. Furey, also assigned to WES from the Computer Sciences Corporation.

At the time of publication of this report, Director of WES was Dr. Robert W. Whalin. Commander was COL Bruce K. Howard, EN.

1 Introduction

For more than twenty years, researchers at the U.S. Army Engineer Waterways Experiment Station (WES) have been performing research dealing with the application of sensors for detection of military targets. The WES research has included the use of seismic, acoustic, magnetic, and other sensors to detect, track, and classify military ground targets. Most of the WES research has been oriented toward the employment of such sensors in a passive mode. Techniques for passive detection are of particular interest to the Army because of certain advantages over active detection. Passive detection methods are not susceptible to interception, detection, electronic/electromagnetic jamming, or location of the source by the threat. A decided advantage for using acoustic and seismic sensors for detection in tactical situations is the non-line-of-sight capability; e.g., detection of low-flying helicopters at long distances without visual contact.

This study was conducted to analyze one method for passive detection, the passive acoustic ranging (PAR) concept, previously developed at WES (Olson and Cress). The analysis used an extensive data set from the Joint Acoustic Propagation Experiment (JAPE).

Background

The PAR concept exploits the repetitive nature of helicopter acoustic signatures to compute the range and velocity of a passing helicopter. The PAR method analyzes the change of the frequency of the signatures through time, known as the Doppler shift, and calculates these parameters by applying the Doppler shift equations to the signals as the helicopter passes through the closest point of approach (CPA). Olson and Cress, 1992[1], present the basic equations governing the extraction of range and velocity for the PAR method. While the PAR method was used on only one set of data, for one helicopter at one site, it produced very promising results. A more comprehensive investigation of the concept was needed, however, to provide a satisfactory evaluation of its performance in typical applications.

[1] Olson, R. E. and Cress, D. H. (1992). "Passive Acoustic Range Estimation of Helicopters," WES Technical Report EL-92-13. USAE Waterways Experiment Station, Vicksburg, MS.

Objective

The objective of this research was to investigate operational constraints on the PAR concept, to estimate the limitations of realistic operation under various environmental conditions, and to determine if the PAR method will apply to multiple helicopters.

2 Analysis of Passive Acoustic Ranging (PAR)

Olson and Cress present the equations governing the behavior of the Doppler shift as a function of velocity and range. The scenario of a helicopter passing an acoustic sensor is shown in Figure 1. The time that the signal is received at the sensor, t_r, is equal to the time that it is emitted, t_e, plus the time of travel of the propagating signal. If the emitting source travels with constant velocity, v, through CPA (see Figure 1) then t_e expressed as a function of t_r is

$$t_e - t_{cpa} = \frac{t_r - t_{cpa} - \sqrt{\beta^2 (t_r - t_{cpa})^2 + (1 - \beta^2)\, \rho^2}}{1 - \beta^2} \tag{1}$$

where

β = speed of the source divided by the speed of propagation

ρ = CPA distance divided by the speed of propagation

t_{cpa} = time when the source passes through CPA

The general Doppler shift formula is

$$f_r = f_e \frac{dt_e}{dt_r} \tag{2}$$

where

f_e = stationary emitted frequency of the source

f_r = frequency received at the sensor

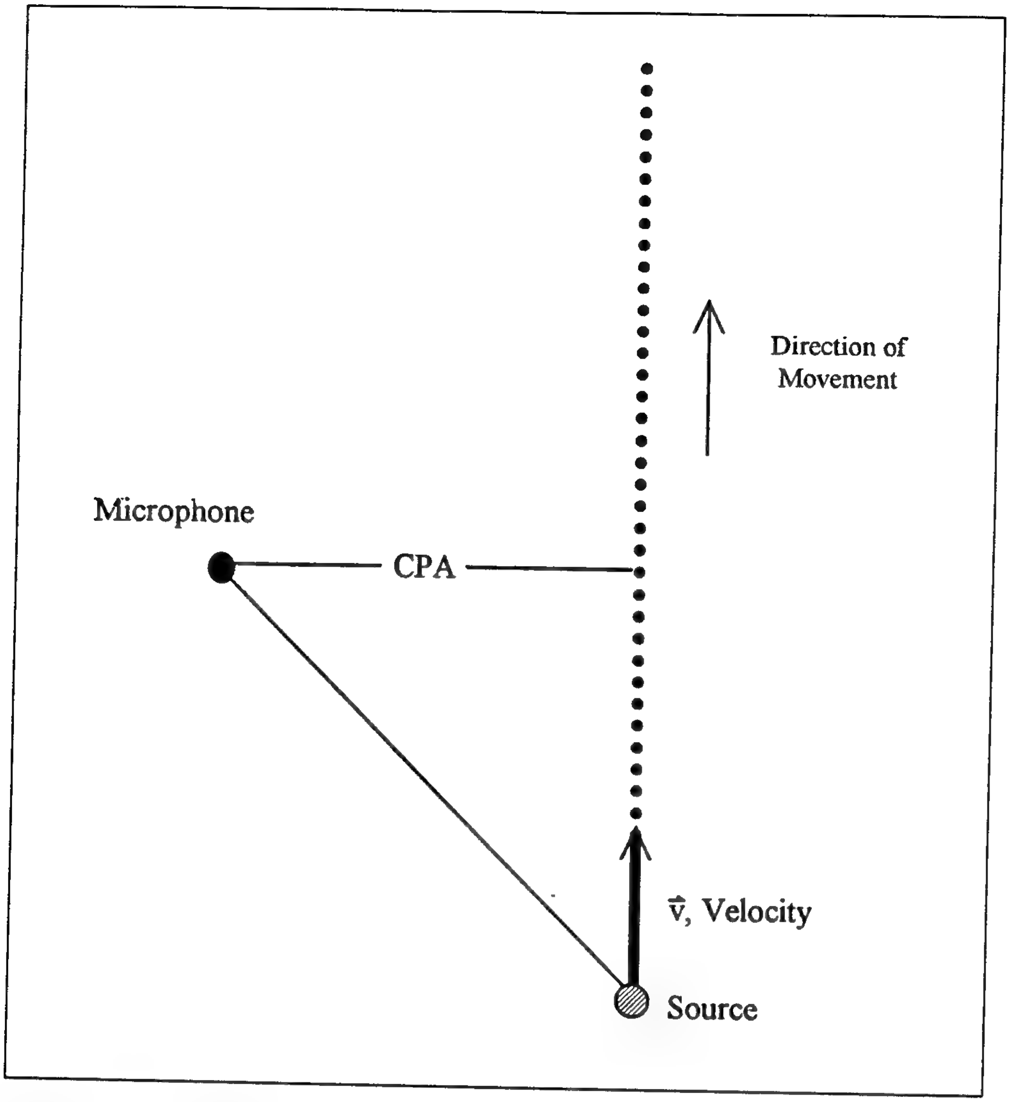


Figure 1. Diagram of source approaching a microphone, showing the closet point of approach (CPA)

For the case under consideration, this formula is expressed in relevant kinematic parameters as

$$\frac{dt_e}{dt_r} = \frac{(1 - \beta\sigma)}{(1 - \beta^2)}$$

where

$$\sigma = \frac{\beta\,(t_r - t_{cpa})}{\sqrt{\beta^2\,(t_r - t_{cpa})^2 + (1 - \beta^2)\,\rho^2}} \qquad (3)$$

The Doppler shift is shown in Figure 2 for various values of these parameters. Note that time is measured from the observed passing of CPA by the vehicle.

Inherent Errors

The basic method for the extraction of target kinematic information (range and speed) from these time traces was to transform the time trace into the frequency domain. The sources had stable characteristic frequencies, which facilitated the extraction of the incoming and outgoing frequencies from the data. The speed and fundamental frequency of the source were then computed, and frequency information as a function of time was obtained from transforms of selected windows of the time trace. As done in the previous study, these data were then used to determine the range by fitting the slope of the frequency versus time curve at CPA to the equation:

$$\frac{df_r}{dt_r} = \frac{-\beta^2 f_e}{\rho} \tag{4}$$

The key to this operation is the transformation from the time to the frequency domain, because of the spreading of the data points through the Doppler shift. Since the slope of the curve is largest at CPA, the spacing of the individual points is sparsest in that region (as can be seen in Figure 3). The accuracy of the slope calculation is dependent on the size of the time step of each window.

The slope at CPA is estimated using times before and after CPA. The range estimate is larger than the true range even for an ideal case such as illustrated in Figure 3. There is an inherent range calculation error associated with the time step. The approach normally used to produce a data set with frequency as a function of time is to extract sections of the time trace and perform a Fast Fourier transform (FFT) on each of these sections. The problem with the FFT method is that, as the length, Δt, of the sections gets smaller, the resolution, Δt, of the frequency grows larger, as seen in Equation 5 below.

$$\Delta f = 1/\Delta t \tag{5}$$

Figure 4 contains graphs of range error as a function of time steps for several values of velocity and range. It can be seen from these graphs that this error is at a minimum when the time step has the smallest value. However, a balance must be drawn between the range calculation error and the frequency resolution of the FFT. Several processing methods were tested in an attempt to overcome the frequency resolution and range error trade-off.

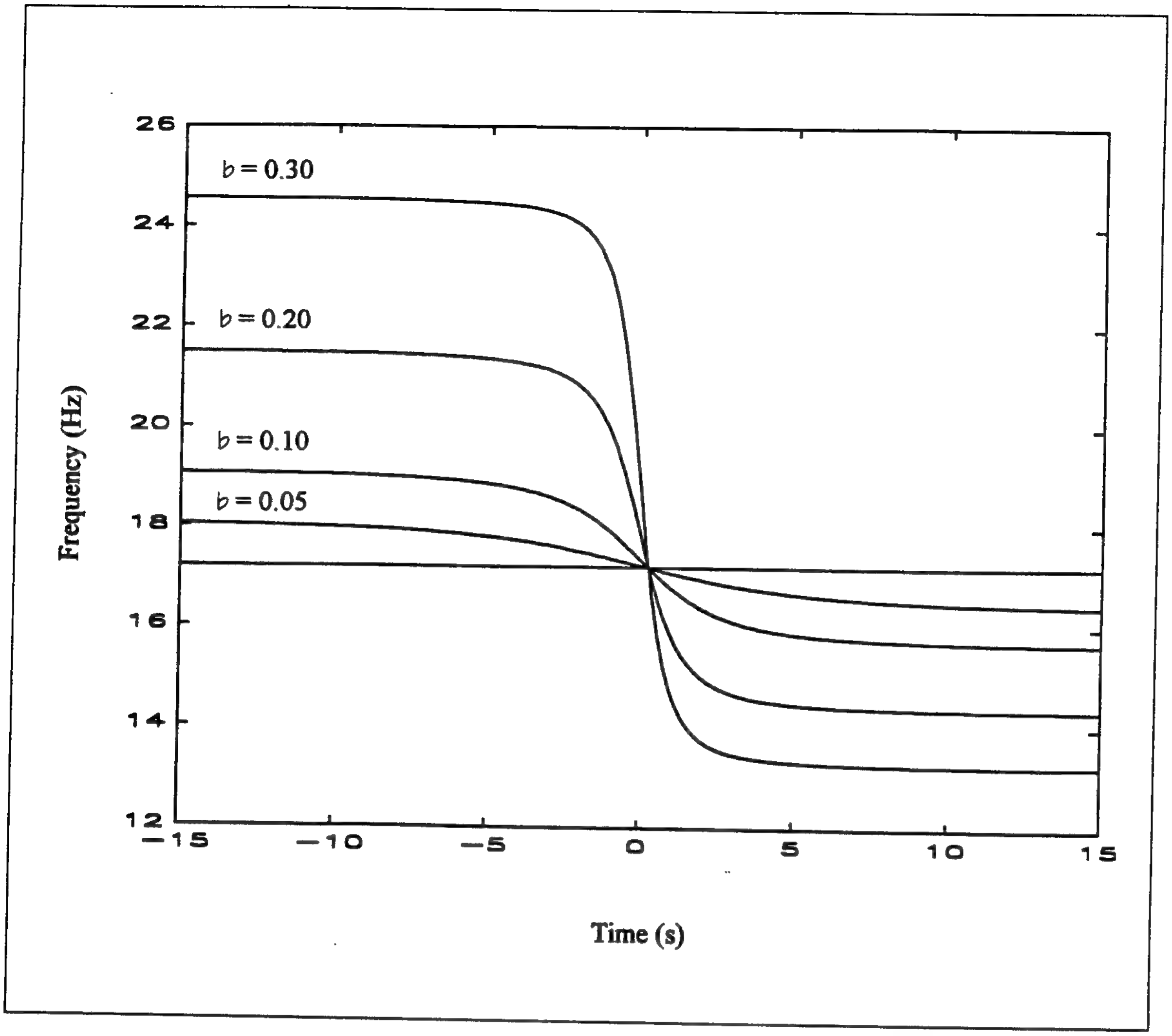


Figure 2. Plot of frequency-vs-time of a source signal with a fundamental frequency of 17.2 Hz at a range of 100 m, for source velocity over sound velocity, beta, of 0.05, 0.1, 0.2, and 0.3

Processing Methods

Simulated helicopter runs (see Appendix A) were developed for the initial evaluation of potential methods. A further evaluation was performed using data from the WES acoustic/seismic database. The two most successful processing methods were the Cross Spectral Density (CSD) method and the method of Successive Differences (SD).

Cross spectral density method

The CSD method, which was initially proposed for this purpose in the previous study, uses a phase analysis of the FFT of the signal to calculate the

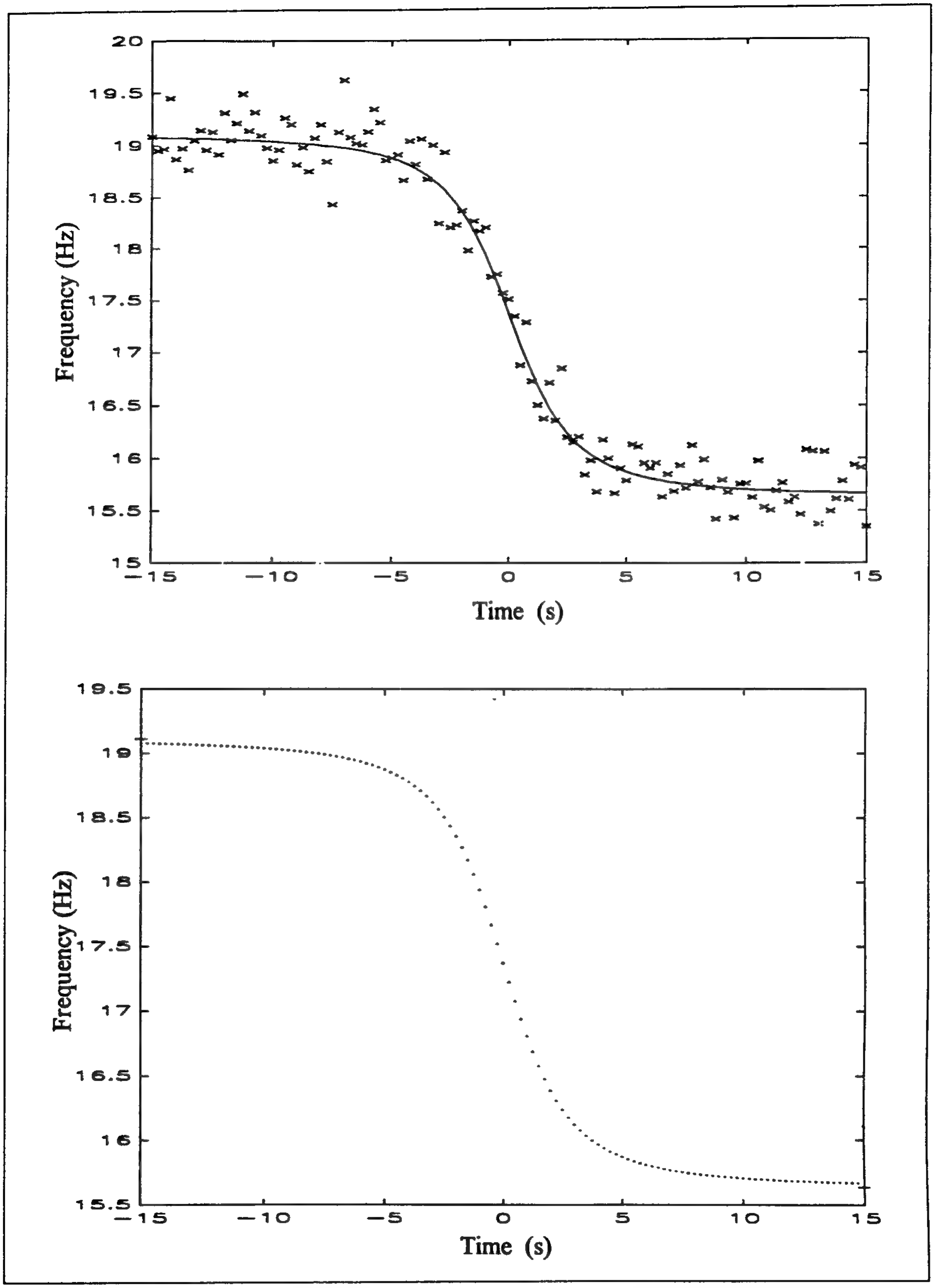


Figure 3. Discrete plot of frequency values for a time step of 0.25 sec with t_{CPA} at $t=0$ seconds. Note the spacing of the points near CPA. The top graph shows real data and curve fit, and the bottom graph shows an example of the ideal case.

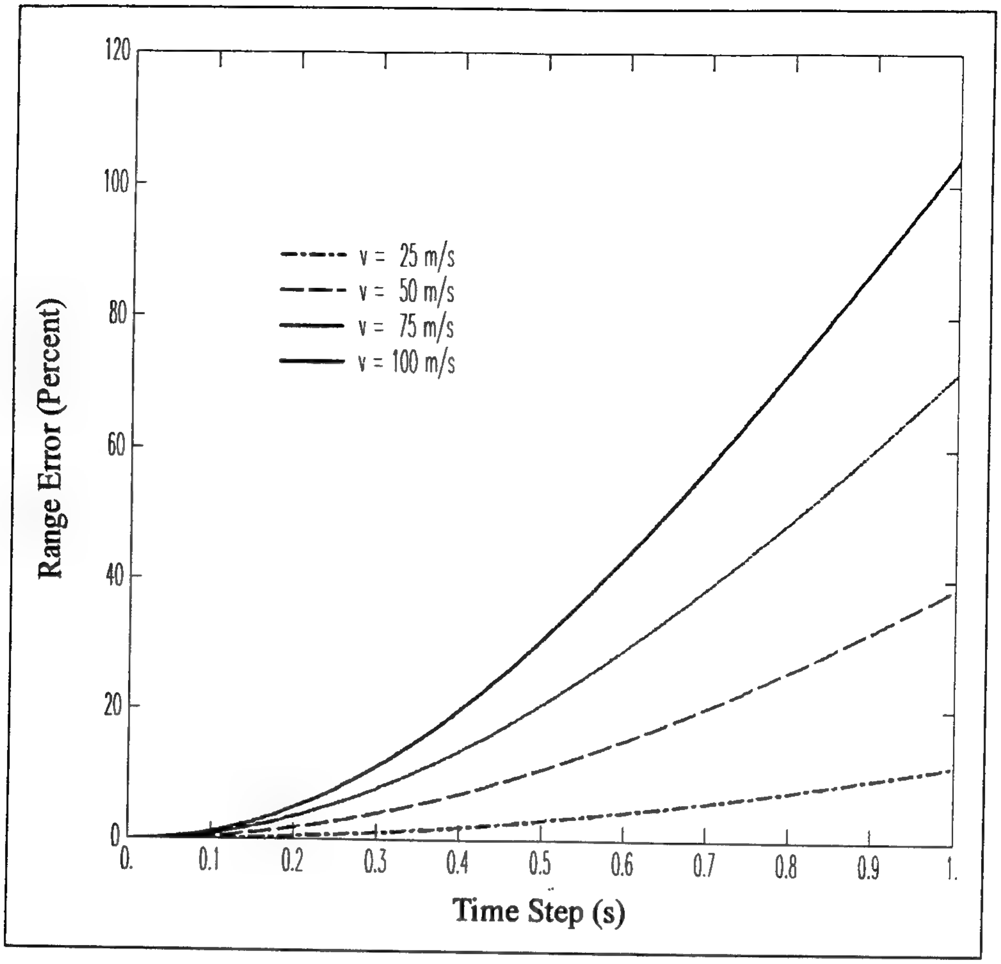


Figure 4. Inherent range error vs duration of time step

frequency of the signal. The FFT's of successive intervals are compared by conjugation and multiplication. For a pure frequency, this method can give accurate results, but because of the nature of the FFT calculation, peaks which are closely spaced in frequency will overlap and interfere in the phase domain. It should be possible to obtain equations which will perform the CSD analysis for peaks containing multiple components, but it is necessary to know beforehand how many components are present. This may not be possible in a field application, where any number of sources may pass the sensor.

The CSD method was tested on the simulated cases listed in Table 1. Further tests were performed using experiment data, and the sources of these data are listed in Table 2. The true values of position as a function of time were used to calculate velocity (actual values were obtained from radar telemetry). Error values (which can be found in Table 3) were within expected limits of the values that were predicted in the graphs in Figure 4.

Table 1
Acoustic Data for a Simulated Helicopter

Runs Generated				
Name of Run	Fundamental Frequency (Hz)	Length of Run (Sec)	CPA (m)	Speed (m/s)
Hela.dat	16	32	10	50
Helb.dat	16	32	50	50
Helc.dat	16	32	200	50
Held.dat	16	32	500	50
Hele.dat	16	32	10	100
Helf.dat	16	32	50	100
Helg.dat	16	32	200	100
Helh.dat	16	32	500	100
Heli.dat	16	32	10	250
Helj.dat	16	32	50	250
Helk.dat	16	32	200	250
Hell.dat	16	32	500	250

the predictions made by the CSD method and error values from these calculations are given in Table 4. Note that the errors stated in these tables are somewhat larger than predicted in Figure 4. This is caused by the addition of background noise to the signal peaks that were used to calculate accurate frequencies.

The CSD allowed an accurate determination of the frequency from a small portion of the spectrum, but the deconvolution of multiple signals using this method had some limitations. The presence of multiple signals in a single peak are not accounted for in this method. The condition of peaks overlapping or not being fully resolved causes the phase information for a particular peak to be corrupted, thus severely complicating the deconvolution of the peak of interest. Acoustic signals for one and two helicopters are shown as an example in Figures 5 and 6

Table 2
Acoustic Data of a Helicopter (Measured)

Case	Run
Case A	WSR147a
Case B	WSR148a
Case C	WSR149a
Case D	WSR150a
Case E	WSR151a
Case F	WSR152a
Case G	WSR156a
Case H	WSR157a

Table 3
Results of CSD Analysis on Multiple Simulated Single Source Signals

Actual Fundamental Frequency (Hz)	Calculated Fundamental Frequency (Hz)	% Error	Actual CPA (m)	Calculated CPA (m)	% Error	Actual Speed (m/s)	Calculated Speed (m/s)	% Error
16	16.002	0.012	10	19.87	98.7	50	49.99	0.02
16	16.004	0.025	50	60.57	21.1	50	49.91	0.18
16	16.017	0.106	200	219.66	9.83	50	48.51	2.98
16	16.047	0.294	500	399.52	20.1	50	41.87	16.26
16	15.998	0.012	10	36.20	262	100	99.94	0.06
16	16.000	0	50	77.16	54.3	100	99.92	0.08
16	16.020	0.125	200	278.81	39.4	100	99.23	0.77
16	16.094	0.587	500	634.80	26.9	100	95.09	4.91
16	16.003	0.019	10	108.34	983.4	250	249.99	0.01
16	16.013	0.081	50	124.01	149.0	250	249.93	0.03
16	16.001	0.006	200	768.65	284.3	250	249.92	0.03
16	16.003	0.019	500	108.34	78.3	250	249.09	0.36
100	100	0	10	0	100	50	49.99	0.02
100	100.008	0.008	50	59.41	18.82	50	49.9	0.20
100	100.101	0.101	200	219.79	9.895	50	48.4	3.20
100	100.289	0.289	500	401.97	19.606	50	41.97	16.1
100	100.002	0.002	10	34.32	243.2	100	99.99	0.01
100	100.007	0.007	50	81.75	63.5	100	99.96	0.04
100	100.125	0.125	200	275.08	37.54	100	99.23	0.77
100	100.566	0.566	500	649.11	29.822	100	95.54	4.463

Note: The 16 Hz componenet corresponds to the main motor fundamental frequency. The 100 Hz component corresponds to the tail rotor. The frequencies were very accurately calculated by this method, so it was necessary to keep many significant figures to show a difference

Table 4
Results of CSD Analysis on Experimental Data

Case	Radar Velocity (m/s)	CSD Velocity (m/s)	Percent Error Velocity	Radar Range (m)	CSD Range (m)	Percent Error Range
A	51.4	56.7	10.3	51.0	41.0	19.6
B	52.0	56.4	8.4	105.4	54.0	48.8
C	54.2	56.3	4.0	105.2	42.0	60.1
D	73.8	71.7	2.8	108.7	99.0	8.9
E	70.9	69.4	2.1	98.2	54.0	45.0
F	75.0	75.2	0.3	109.3	62.0	43.3
G	47.1	57.1	21.1	206.4	141.0	31.7
H	44.8	49.6	10.7	320.5	187.0	41.7

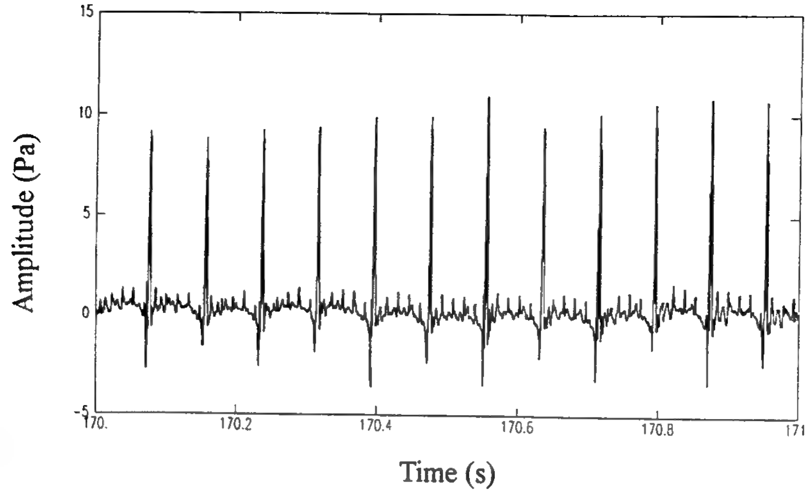
with the power spectral density (PSD) obtained from the FFT. The two helicopters are not resolved in the frequency peaks.

Method of successive differences

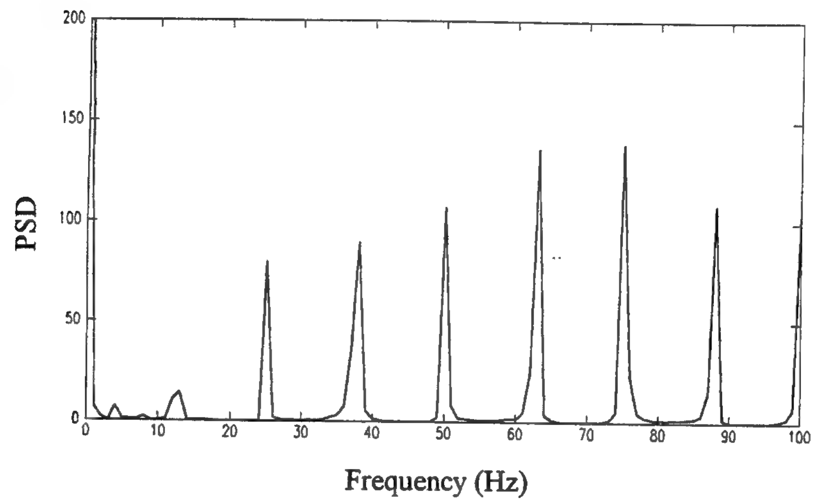
Another way of extracting the frequency and velocity is to first perform an FFT on the entire sample of data (all the way through the approach and departure of the source). The results of the FFTs are shown in Figures 7 - 10. The maxima on the ends of the u-shaped features in these figures are the incoming and outgoing frequency values of the source. It is possible to derive an analytic expression for the u-shaped features in the ideal case.

The problem of extracting frequency as a function of time was addressed using the method of SD. The frequency resolution limit from Equation 5 was overcome by computing the FFT of two intervals with sufficient length to produce the desired resolution. The successive intervals for each FFT had starting points separated by the desired time step. The differences between these two spectra were computed, and the remaining positive portion of the difference contains the peak values of the frequencies that were added into the signal during that time interval.

The SD method exploits the fact that, as the period in the sample to be transformed is increased, the resolution of the result will also increase. The differences between two amplitude-normalized, high-resolution FFTs will represent the changes that occurred during the period between the acquisition of the two time traces.



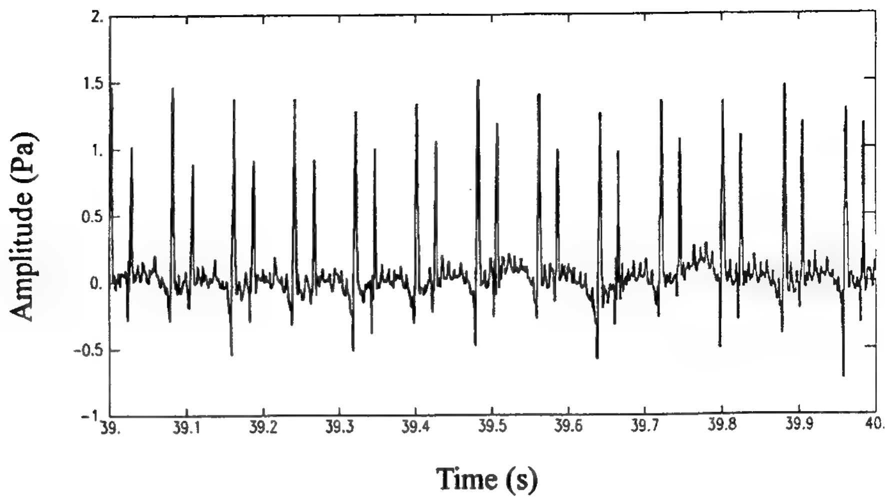
a. Time history



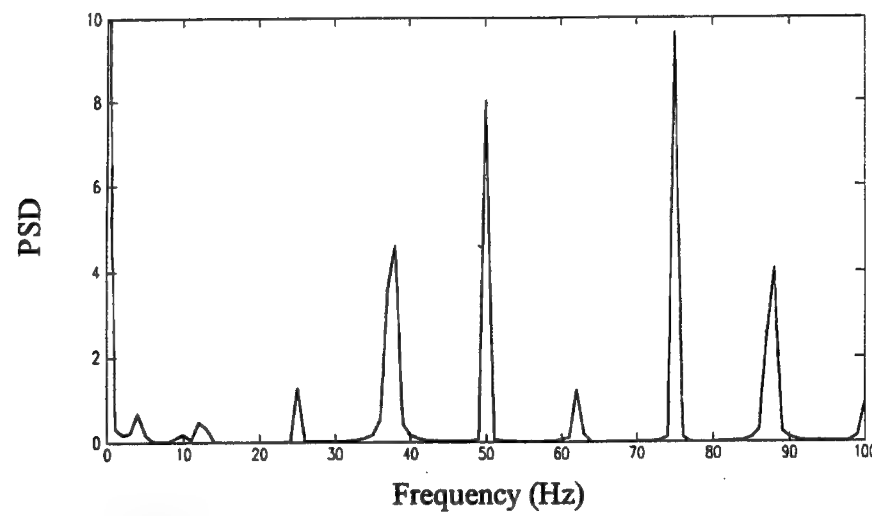
b. Power spectral densities (PSD)

Figure 5. Signals from one helicopter

By first determining the incoming and outgoing frequencies, the stationary fundamental frequency of the source was calculated. From this information, the time trace was then searched at a high resolution in order to determine the slope of the frequency-versus-time function as the fundamental peak passed through the stationary value. This process was automated along with peak search algorithms to identify peaks and extract frequency information from them. The values of range, velocity, and frequency were the output of this code. Errors in output from this method agree with what is expected from the theoretical limits.



a. Time history



b. Power spectral densities (PSD)

Figure 6. Signals from two helicopters

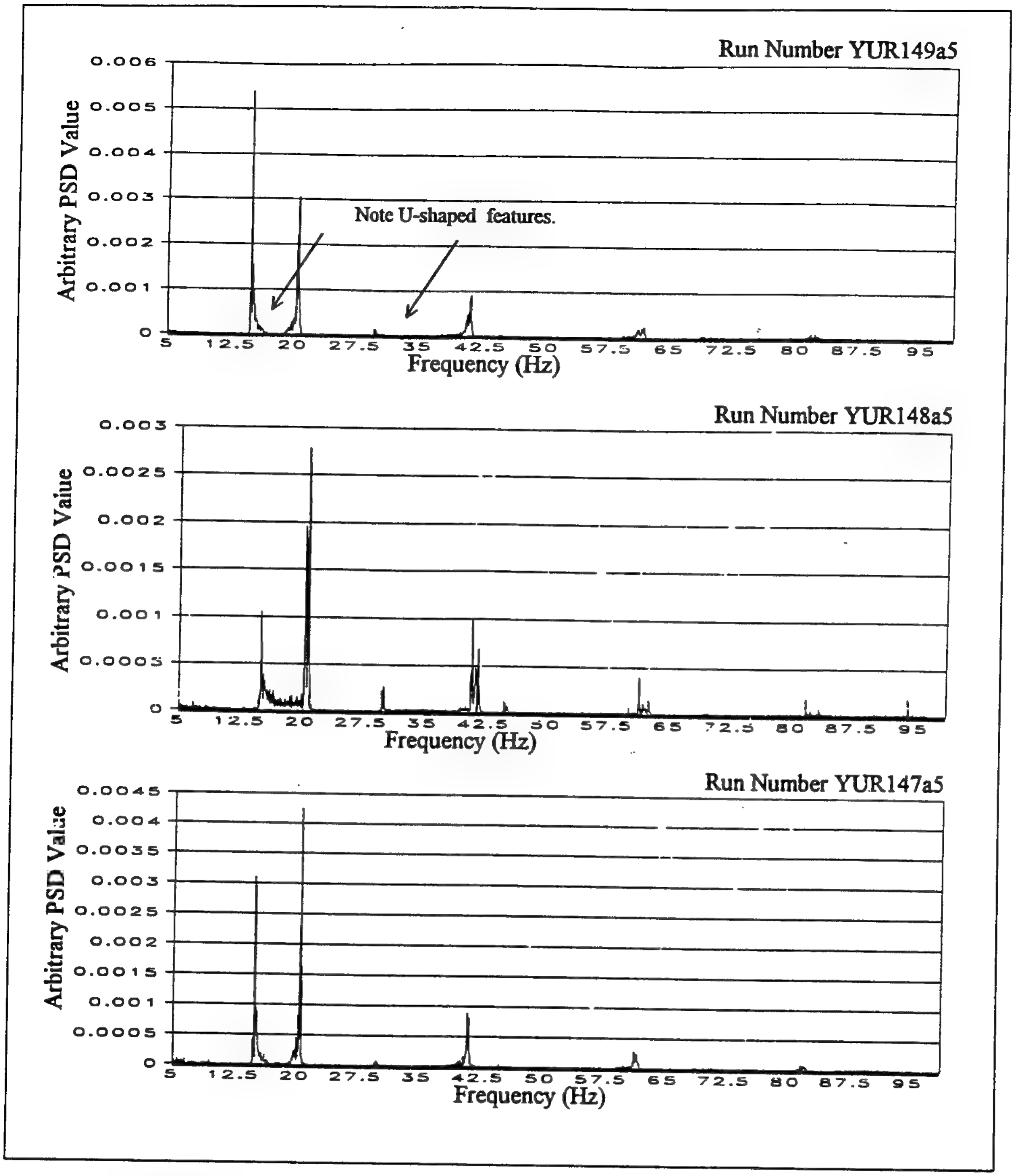


Figure 7. Selected PSD plots for 30-second extractions of time traces about CPA

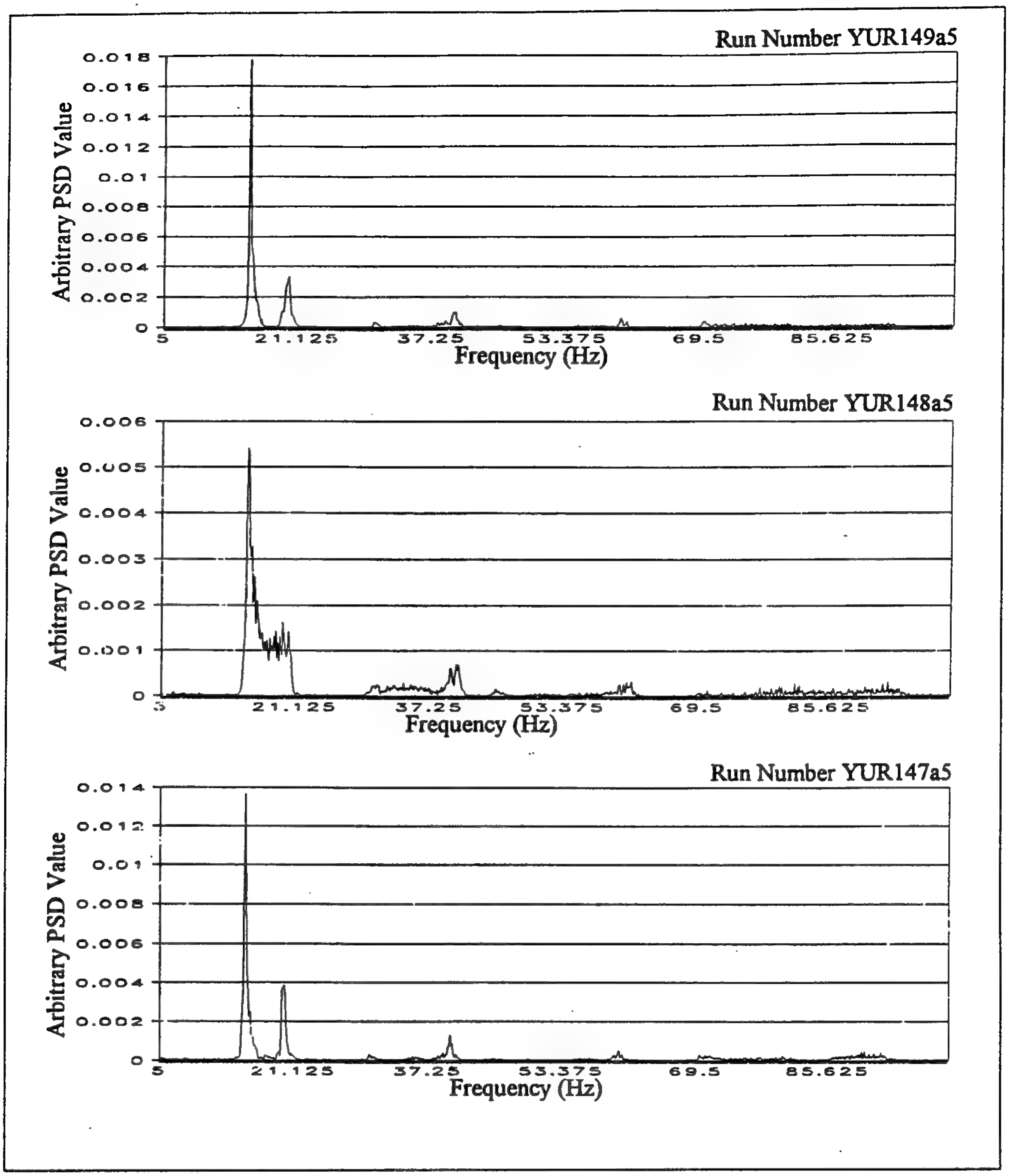


Figure 8. Selected PSD plots for 15-second extractions of time traces about CPA

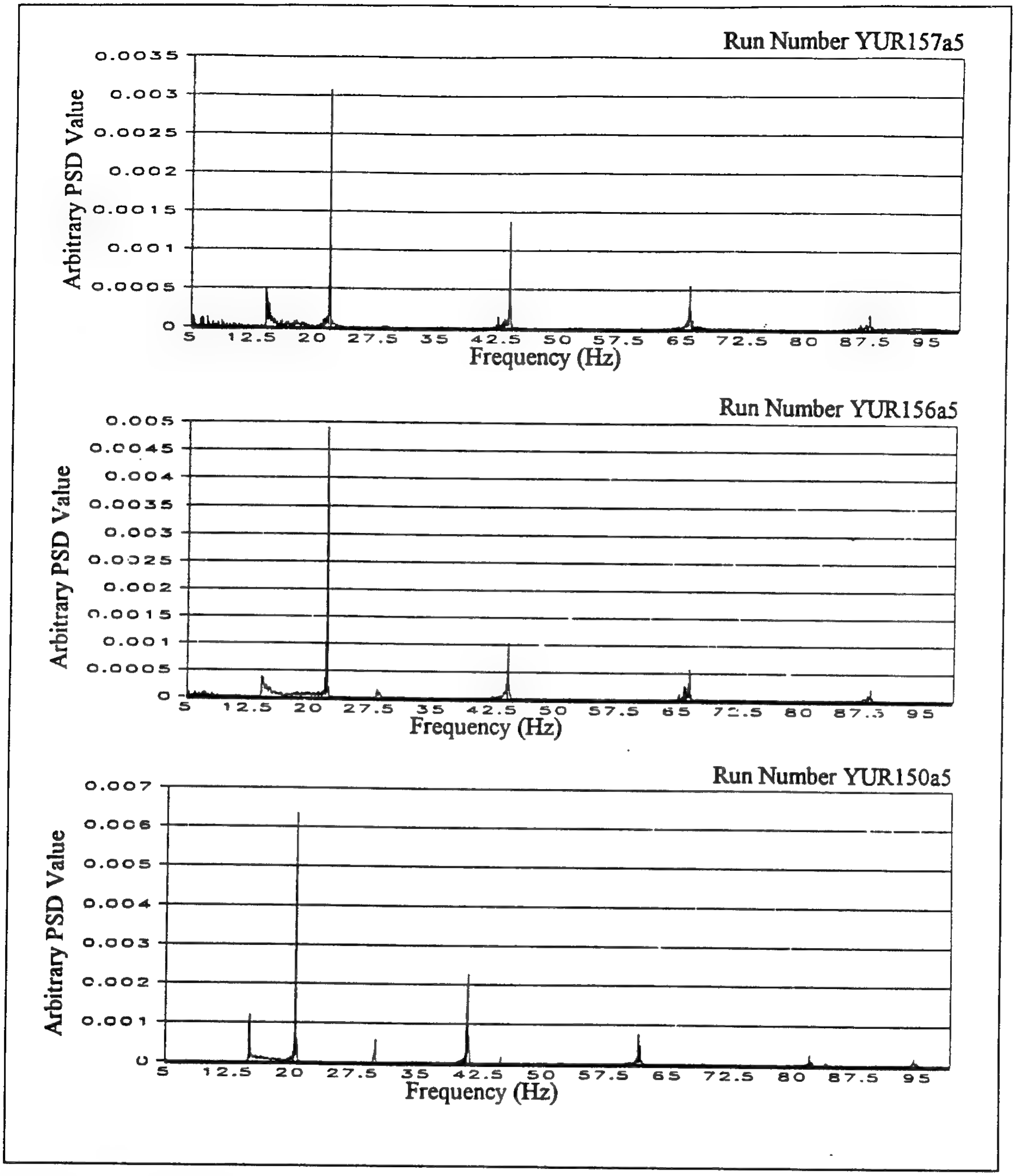


Figure 9. Selected PSD plots for 30-second extractions of time traces about CPA

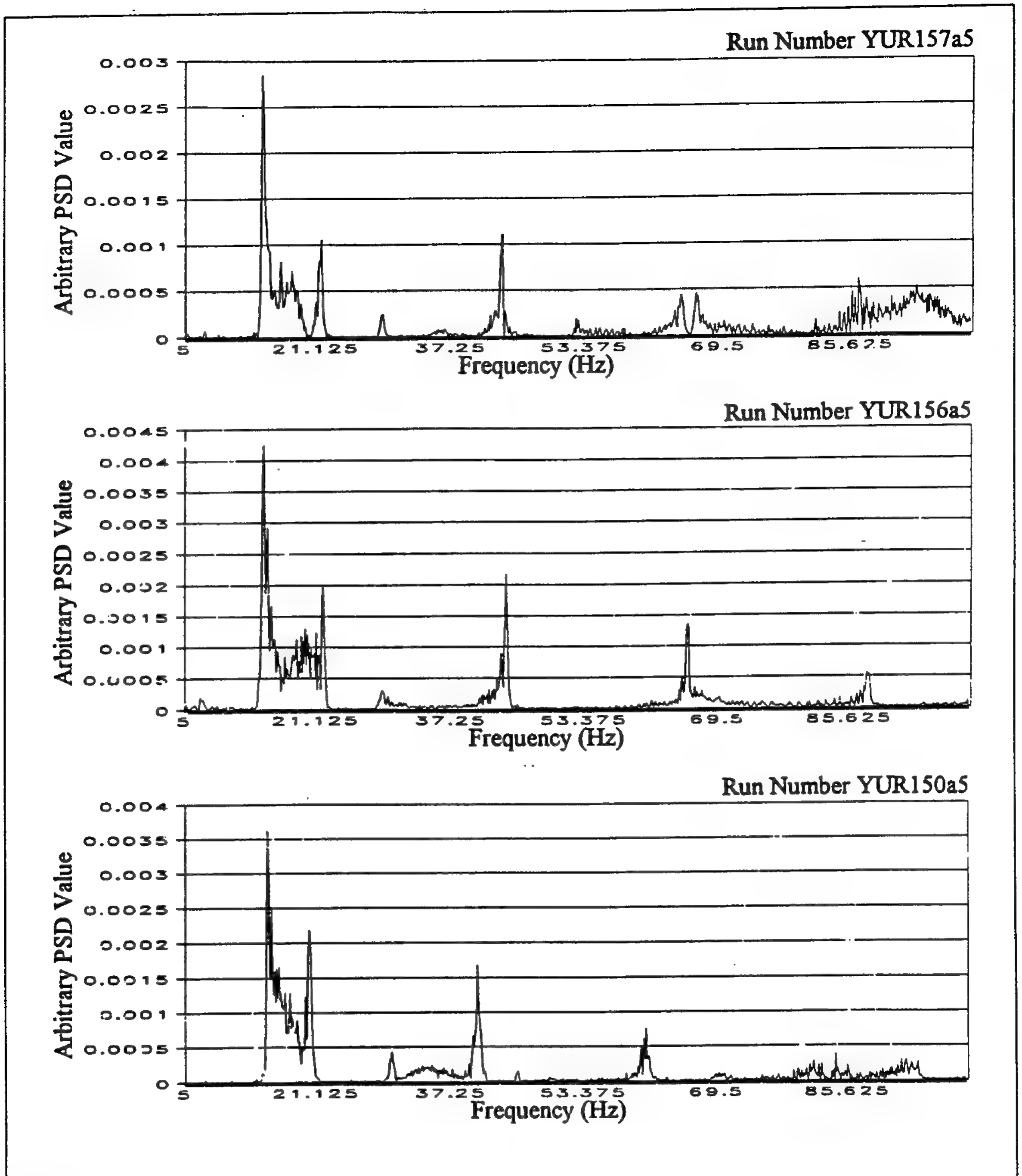


Figure 10. Selected PSD plots for 15-second extractions of time traces about CPA

Data processed with the CSD method were also analyzed using the SD method. The results of the simulated and experimental runs, mentioned previously in Tables 1 and 2, are presented in Tables 5 and 6.

Comparison of methods

While both methods produced results which agree with the theoretical analysis of the error in the ideal case, the SD method had an advantage over the CSD when there were multiple signals within a single peak. In addition to the multiple peak problem, the CSD displayed a loss of accuracy when dealing with sections of a time trace that had a duration of less that 0.25 second. The SD has no such limit. The disadvantage of the SD method is the intensive calculations that must be performed. The SD algorithm, written in the C programming language and executed on a 80486-class, 33-MHz personal computer, took longer to run. The purpose in this analysis was not to develop a fast code, but to perform a comprehensive evaluation of the two methods. Speed limitations of the SD method could be overcome by making further improvements in the code.

Both the CSD and SD methods were used with the equations from the previous study[1] to calculate range, speed, and frequency information for a single source, with some limited success. The problem of deconvoluting the signal was not solved at this time, but the key to developing a system for obtaining range information from multiple sources is to identify the number of sources present, and to track the individual signature peaks through their stationary frequencies.

Refinement of PAR through Artificial Neural Networks

The Artificial Neural Network (ANN) is a computer algorithm that is modeled after the synapses and neurons in the brain. Recent research into the applications of this technique have found great success in the classification and prediction of non-linear data. Investigations have been made using an ANN in a wide variety of fields and applications, ranging from classification of military targets to medical diagnosis. The studies show that an ANN has the capability to learn a non-linear pattern. Consequently, a study was conducted to evaluate the feasibility of using an ANN to classify the source type and possibly to determine the number of sources, so that the PAR method can compute the velocity and range of each source.

[1] Olson, R. E. and Cress, D. H. (1992). "Passive Acoustic Range Estimation of Helicopters," WES Technical Report EL-92-13. USAE Waterways Experiment Station, Vicksburg, MS.

Table 5
Results of SD Analysis on Multiple Simulated Single Source Signals

Actual Fundamental Frequency (Hz)	Calculated Fundamental Frequency (Hz)	Error %	Actual CPA (m)	Calculated CPA (m)	Error %	Actual Speed (m/s)	Calcu-lated Speed (m/s)	Error %
16.00	16.03	0.19	50	49.75	0.50	10.00	22.00	120.00
16.00	16.03	0.16	50	47.88	4.25	50.00	40.00	20.00
16.00	16.02	0.12	50	40.90	18.21	200.00	99.00	50.50
16.00	16.19	1.19	50	25.68	48.65	500.00	64.00	87.20
16.00	15.98	0.10	100	100.02	0.02	10.00	48.00	380.00
16.00	16.02	0.12	100	98.79	1.21	50.00	68.00	36.00
16.00	16.16	1.00	100	93.88	6.12	200.00	117.00	41.50
16.00	16.26	1.60	100	79.83	20.17	500.00	194.00	61.20
16.00	15.99	0.05	250	250.03	0.01	10.00	122.00	1120.00
16.00	15.99	0.05	250	250.03	0.01	50.00	124.00	148.00
16.00	16.17	1.08	250	248.68	0.53	200.00	240.00	20.00
16.00	16.61	3.79	250	245.00	2.00	500.00	441.00	11.80

Table 6
Results of SD Analysis on Experimental Data

Case	Radar Velocity (m/s)	CSD Velocity (m/s)	Percent Error in Velocity	Radar Range (m)	CSD Range (m)	Percent Error in Range
A	51.4	55.8	8.5	51.0	45.8	10.2
B	52.0	53.8	3.4	105.4	69.7	33.9
C	54.2	52.4	3.2	105.2	72.6	31.0
D	73.8	72.9	1.2	108.7	99.0	8.9
E	70.9	71.1	0.3	98.2	78.3	20.3
F	75.0	74.8	0.3	109.3	92.4	15.5
G	47.1	52.1	10.5	206.4	141.0	31.7
H	44.8	47.8	6.7	320.5	243.5	24.0

Back-propagation training of an ANN

An ANN is an interconnected array of neurons. The first level of neurons, called the input layer, is connected to hidden internal layers of neurons by synapses. The last hidden layer of neurons is also connected to the output layer by synapses. The neurons in the hidden layers are made up of weight values, which are established by training. The ANN is trained to recognize patterns in the data by learning from many examples. The standard method of training is by "back-propagation". A transfer function for each neuron determines the value that neuron will output. Many possible modifications can be made to optimize the internal parameters of an ANN.

ANN training, testing, and development

An ANN application was designed for the helicopter scenario. The standard method of back-propagation was used for training in this case. The program for generating an ANN training set is presented in Appendix B, and the ANN back-propagation code is presented in Appendix A. The standard sigmoid transfer function, and an input, output, and single hidden layer of neurons, were chosen for the architecture of this network for this application.[1] The ANN output features are normalized to 1 or 0, depending on the presence or absence of that feature in the input. Since there were only single runs of Huey and Blackhawk helicopters (no multiple runs) in the JAPE, a method to synthesize signals from more than one source was developed. A simulation code (further discussed in Appendix B) was also used to create signals from a third source. Data from each source were used in training the ANN.

The first ANN was trained on three data sets (listed in Table 7), which included a single simulated source, a single helicopter (run WS147 from the JAPE), and a dual source synthesized from these two runs. A total of 350 one-second time windows were extracted at random from the time traces of these three data. Fifty of these were retained for testing of the ANN. An FFT was performed on each of the 350 time windows, and the results were truncated to limit the frequency content to a range of 5 to 85 Hz. These frequency values were used as the input layer for the ANN. The ANN was then trained on the FFT of each of the first 300 time windows until the average error was minimized. After the training was complete, the 50 one-second windows retained for testing were fed into the ANN as inputs. The ANN was able to classify the source accurately 100 percent of the time, as indicated in Table 8.

---

[1] Hecht-Nielsen, R. (1991). *Neurocomputing*. Addison-Wesley Publishing Company, New York.

Table 7
Index of Cases Used in First ANN Training

Case A	Simulated Source with 16Hz Fundamental
Case B	WSR147a4 Blackhawk Helicopter from JAPE Data Set
Case C	Superposition of Cases A and B

Table 8
Results of ANN Classification of First Training Exercise (Single Helicopter Sources)

Case A		Case B		Case C		
Predicted	Actual	Predicted	Actual	Predicted	Actual	Results
0.00	0.00	0.00	0.00	0.99	1.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.03	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.99	1.00	0.01	0.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	0.02	0.00	1.00	1.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.01	0.00	0.00	0.00	1.00	1.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct

(Continued)

Table 8
Results of ANN Classification of First Training Exercise (Single Helicopter Sources)

Case A		Case B		Case C		
Predicted	Actual	Predicted	Actual	Predicted	Actual	Results
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	0.08	0.00	0.99	1.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.98	1.00	0.06	0.00	0.01	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.81	1.00	0.01	0.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
1.00	1.00	0.04	0.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.97	1.00	0.18	0.00	0.00	0.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
1.00	1.00	0.00	0.00	0.00	0.00	Correct
0.98	1.00	0.16	0.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	1.00	1.00	0.00	0.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	0.00	0.00	1.00	1.00	Correct
0.00	0.00	0.02	0.00	1.00	1.00	Correct

After this initial success, nine data sets were prepared on single and combined sources, as shown in Table 9. A total of 550 time windows were selected from these data sets, and 50 of these were again retained for testing. Another ANN was trained on the FFT of each of the first 500 time windows to determine if the ANN could classify multiple helicopters. For the 50 retained time windows, the ANN was able to classify the sources accurately 90 percent of the time in any of the combinations. The results of the testing are presented in Table 10 in which the output of the ANN is presented as a number from 0 to 1.0 for each specific case (Note that the cases are defined in Table 9). A maximum value for a specific case indicates that the ANN identified the signal as originating from that case. The truth values (presented in the same format) represent the actual source of the signal. Selected FFT's from the test cases are presented in Appendix C.

ANN Peak Extraction

After achieving success in classifying the sources of the signal, the next step was to extract peak signal information and identify it with a particular source. This allows the ANN to build the frequency-versus-time curve for each source. The curve can then be used to determine the source range and velocity. The approach was to use a training set for the ANN that contained FFT data from two successive time intervals. For these time intervals, the true frequency for a given time was calculated using the Doppler shift equations, with data for position and velocity obtained from radar tracking information.

Ten separate training runs were prepared and attempted using the WES supercomputer, but these runs produced little success. While the ANN would converge on the training set, it would fail on the testing set. Each of the test sets would converge to a particular frequency value. After investigating further, it was determined that these results were produced by the uncertainty in the position and velocity data obtained from the radar measurements. Further uncertainties are produced in the calculation itself because of the nature of Equation 1.

Table 9
Index of Cases for Artificial Neural Net Classification

CASE	Description
Case A	Simulated Helicopter
Case B	Huey from Data Set WSMR010
Case C	Blackhawk from Data Set WS147
Case D	Two Summed Case A
Case E	Two Summed Case A and Case B
Case F	Two Summed Case A and Case C
Case G	Two Summed Case B
Case H	Two Summed Case B and Case C
Case I	Two Summed Case C

Table 10
Results of ANN Classification of Second Training Exercise (Multiple Helicopters)

	Case A	Case B	Case C	Case D	Case E	Case F	Case G	Case H	Case I	Results
output	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	
truth	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.95	0.00	0.00	0.00	0.00	0.39	0.00	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.01	
truth	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	Correct
output	0.00	0.92	0.00	0.00	0.00	0.00	0.04	0.02	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.01	
truth	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	Correct
output	0.00	1.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	
truth	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.00	0.46	0.00	0.00	0.00	0.00	0.04	0.00	
truth	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.99	0.00	0.00	0.00	0.00	0.03	0.00	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.01	
truth	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.01	
truth	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	Correct
output	0.00	0.28	0.00	0.00	0.00	0.00	0.93	0.00	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Incorrect
output	0.00	0.00	0.11	0.00	0.00	1.00	0.00	0.00	0.00	
truth	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	
truth	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.97	
truth	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	Correct
output	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.01	
truth	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	Correct
output	0.00	0.00	0.31	0.00	0.66	0.00	0.00	0.00	0.00	
truth	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	Incorrect
output	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
truth	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Correct

(Sheet 1 of 3)

Table 10 (Continued)										
	Case A	Case B	Case C	Case D	Case E	Case F	Case G	Case H	Case I	Results
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.60 1.00	1.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.99 1.00	0.02 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.79 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	1.00 1.00	0.01 0.00	Correct
output truth	0.00 0.00	0.01 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	1.00 1.00	0.01 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.13 0.00	0.00 0.00	0.00 0.00	0.98 1.00	Correct
output truth	0.00 0.00	0.03 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.96 0.00	0.01 0.00	0.00 0.00	Correct
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.05 0.00	0.67 1.00	Incorrect
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.07 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.08 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	1.00 1.00	0.01 0.00	Correct

(Sheet 2 of 3)

Table 10 (Concluded)										
	Case A	Case B	Case C	Case D	Case E	Case F	Case G	Case H	Case I	Results
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.88 1.00	0.07 0.00	Correct
output truth	0.00 0.00	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.99 0.00	0.48 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Incorrect
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.09 0.00	0.64 1.00	Correct
output truth	0.00 0.00	0.00 0.00	0.98 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.99 1.00	0.01 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.99 1.00	0.01 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.67 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.01 1.00	0.00 0.00	Correct
output truth	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	0.00 0.00	1.00 1.00	0.00 0.00	0.00 0.00	Correct

(Sheet 3 of 3)

3 Discussion and Conclusions

It appears feasible to develop the PAR concept into an algorithm that can be part of an operational sensor system. However, this study identified several PAR limitations that must be addressed before using such an algorithm. In the previous study[1] the PAR concept was described as operable on a single sensor. However, this study has revealed that the selection of the time step for developing the Doppler shift curve is critical, depending on the speed of the source and the distance from the sensor. Selection of the wrong time step could cause the curve to have only one or two points during the Doppler shift, which would severely distort the slope calculation. Because of this limitation, it would be advisable to use the PAR concept with multiple sensors. An array is not required (i.e., for beamforming), and the PAR algorithm can be used independently on each sensor (which can be separated by relatively large distances).

The PAR concept can also be used on multiple sources, but the Doppler shift curves for each of the sources present in the signal must be identified. The conventional methods of signal processing tested here were not sufficient to reliably produce the curves for each source.

Another limitation of the PAR concept is that the source has to pass through CPA and into the far-field before the calculation can be completed unless there is *a priori* knowledge of the frequency of the source when it is stationary.

The use of an ANN to identify the sources and separate their signal contributions is feasible. However, only a limited test of the ANN for this purpose was performed in this study. It should be noted that the 90-percent accuracy (10-percent error) resulting from use of an ANN is probably not a major problem for application to the PAR concept. The ANN was not optimized, and was only trained with 500 samples. Training on additional samples would increase the accuracy.

---

[1] Olson, R. E. and Cress, D. H. 1992. "Passive Acoustic Range Estimation of Helicopters," WES Technical Report EL-92-13. U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

The accuracy of the ANN classification was also affected by the distance of the source from the sensor, because the signal degraded as it propagated through the atmosphere. The time windows were selected for distances up to 1.5 km from CPA, and the Doppler shift occurs within a few hundred meters of CPA. Therefore, the accuracy of the ANN classifier is higher in the area required for the analysis for the PAR method. The use of an ANN could also improve the response time of the PAR concept. Because of the capability of an ANN to identify the source, the calculation could be performed within a very short time of the source passing CPA.

4 Recommendations

The PAR method has definite utility in the acoustic detection field, but because of the limitations discussed in this report, caution must be used in planning the deployment of a system with a PAR algorithm. A PAR algorithm might be more useful as a supplement to other acoustic detection algorithms (e.g., beamformers used in triangulation).

The application of an ANN to the PAR concept using multiple sensors appears to have the necessary processing capability to overcome the limitations mentioned above. The ability to classify and separate the sources in the measured signal allows the PAR algorithm to determine the stationary frequency and calculate the range and velocity of each source. These calculations can only be done as the source passes through CPA without additional information. A sequence of ANNs could also possibly be used to perform several portions of the signal processing required for the PAR concept. Further study of the PAR concept and the applicable ANN processing is recommended.

The WES Acoustic/Seismic Research Team has recently acquired a Global Positioning System (GPS) system that will allow for high resolution of target position data at any altitude. Further evaluation of the PAR concept using such a high-resolution target location system is recommended.

Appendix A
ANN Computer Code

The following is a listing of the C language code BINGEN2.C; this code was used to preprocess data for use in the training of the ANN. This codes takes as command line input the following:

binary data file name
the output file name for the PSD values
the second to start the PSD,
the number of seconds to PSD,
the sample rate of the input file,
the frequency of the PSD values to start to output,
the frequency of the last PSD values to output,
1 for training set, 0 for testing set
n, the number of outputs for the training set
value for output 1
.
.
value for output n.

The output of this code goes into the output file that was specified and consists of a single line of data which first contains the normalized PSD amplitudes for the frequencies that were specified and, for a training set, the output values for the neural net. This data was then used with the program DELTA32.C to train an ANN. To enter this data from the command line by hand was quite time consuming, so a Quick Basic program, GenNET.bas, was written to produce a batch file that will execute this code for the large number of samples required to build a training set. This program is listed following BINGEN2.C and the subroutines that it calls.

```
BINGEN2.C
#include "define.h"

main( int argc, char *argv[])
{
   float   dummy,value,second,num_seconds,normval,out[20];
   long    dcount,ocount,start_point,nnn,points,maxnnn,pd;
   char    outname[40],inname[40];
   double  maxval;
   int     numouts,trn_or_tst;
   short   x, y, maxamp;

/*****************************************************************
THE FOLLOWING LINES ARE FOR GETTING AND SETTING UP THE INPUT PARAMETERS
GIVEN ON THE COMMAND LINE
*****************************************************************/
```

```
numouts = atoi(argv[9]);
if(argc != 10 + numouts)
{
printf("\nCommand line format is as follows:");
/*  0    1    2    3    4     5   6     7     8     9     10
11    9+numouts  */
printf("\n\n netgen INFILE1 OUTFILE SECOND NUM_SECONDS SRATE FFT_START
FFT_STOP trn_or_tst numouts out0 out1...out[numouts]");
printf("\n  trn_or_tst = 0 for test.");
printf("\n  trn_or_tst = 1 for train.");
exit(0);
}
for(ocount=0;ocount<40;ocount++)
{
  inname[ocount] = argv[1][ocount];
  outname[ocount] = argv[2][ocount];
}

printf("\n inname = %s",inname);
printf("\n outname  = %s",outname);

second      = atof(argv[3]);
num_seconds = atof(argv[4]);
srate       = atof(argv[5]);
fft_start   = atof(argv[6]);
fft_stop    = atof(argv[7]);
trn_or_tst  = atoi(argv[8]);
for(ocount=0;ocount<numouts;ocount++)
  out[ocount] = atof(argv[10+ocount]);

start_point = (long)(second * srate);
points = (long)(num_seconds * srate);

dcount = get_secs(inname, data_array,start_point,points);
for (x=0;x<num_seconds*4;x++)
{
  maxamp=1;
  for (y=x;y<(srate/4);y++)
    if (abs(data_array[y])>maxamp)
    maxamp=abs(data_array[y]);

  for (y=x;y<(srate/4);y++)
    data_array[y] = 32000*(data_array[y]/maxamp);
}
if( (dcount < 1) || (dcount>BUFSIZE) )
{
  printf("\n ERROR dcount = %ld",dcount);
  exit(0);
}
else
{
  printf("\n time length of run = %lf",(double)(dcount)/srate);
  printf("\n points = %ld",points);
}

littlefft(data_array, points);

normval = 0.0;

outfile = fopen(outname,"at");
if(outfile == NULL)
{
```

```
    printf("\nERROR opening %s",outname);
    exit(0);
    }
    maxval = 0.0;
/*   NORMAL OUTPUT */
    for(nnn=(num_seconds*fft_start);nnn<(num_seconds*fft_stop);nnn++)
    {
      fprintf(outfile," %lf ",data_array[nnn]);
      if(data_array[nnn] > maxval)
      {
      maxval = data_array[nnn];
      maxnnn = nnn;
      }
    }
/* END OF NORMAL OUTPUT */

/* OUTPUT FOR QPRO
    for(nnn=(num_seconds*fft_start);nnn<(num_seconds*fft_stop);nnn++)
    {
      fprintf(outfile,"\n %ld %lf ",nnn,data_array[nnn]);
      if(data_array[nnn] > maxval)
      {
      maxval = data_array[nnn];
      maxnnn = nnn;
      }
      }
 END OF OUTPUT FOR QPRO */

 if(trn_or_tst == 0) /* 1 = test */
 {
      for(ocount=0;ocount<numouts;ocount++)
      fprintf(outfile,"  %f ",out[ocount]);
      fprintf(outfile," \n");
 }
}
```

This routine is a called by bingen2.c and extracts the section of the time trace that is necessary to produce the PSD from the binary data file.

```
GET_BIN.c
#include "declare.h"
long get_secs(char filename[], double getarray[],long start_point,long points)
{
   int    eof1;
   float  dummy,value;
   long   dcount,shift;
   short  read_array[1500000];

   if(points > 1500000)
   {
     printf("\n ERROR read_array must be increased to handle points value of %ld",points);
     pause(points);
   }
   printf("\n start_point = %ld   points = %ld",start_point,points);
   infile = fopen (filename,"rb");
   if(infile != NULL)
     printf("\n Opened file %s ",filename);
   else
   {
     printf("\nCouldn't open file %s",filename);
     exit(0);
   }
```

```
    dcount = fseek(infile, start_point, SEEK_SET);
    if(eof1 == EOF)
    {
      printf("\nERROR reached EOF before coming to startpoint.");
      printf("\n    on file %s",filename);
      pause(0);
    }
    printf("\n skipped to start_point %ld in file %s",start_point,filename);

    dcount    = 0;
    dcount=fread(read_array,sizeof(short),points,infile);
    if(dcount != points)
    {
      printf("\nERROR reading from %s ",filename);
      printf("\n only read in %ld points",dcount);
      pause(0);
    }
    if(dcount >= BUFSIZE)
    {
      printf("\n MUST INCREASE ARRAY SIZE");
      exit(0);
    }
    for(shift=0;shift<dcount;shift++)
      getarray[shift] = read_array[shift];
    fclose(infile);
    return dcount;
}
```

This routine is called by BINGEN2.C to calculate the PSD of the data that is passed by GETBIN.C. It is and adaptation of an algorythm found in *** and was coded and tested by Travis Harrell and Cliff Morgan.

```
/*****************
*****************/

#include "declare.h"
#include "buffers.h"

void littlefft(littlearray,npt)
double littlearray[];
long npt;
{
   int     datacounter,eof1,f;
   int     mr,nn,l,istep,el,pstart,pstop;
   long    i,m,j,replace;
   double  tr,ti,a,wr,wi,fft_time;
   double  fftval,data1;
   float   stepsize,step;
   double  temparray[18000];

   fft_time = (double)(npt) / srate;
   if(npt >= BUFSIZE)
   {
     printf("\n ERROR must increase array size. ");
     exit(0);
   }

/*
   Do the fft calculations.
*/
```

```
mr  = 0;
nn  = npt - 1;

for(i=1;i<=npt;i++)
{
  fr[i] = 0.0;
  fi[i] = 0.0;
}
for(m=1;m<=npt;m++)
  fr[m] = (littlearray[m-1]);

for(m=1;m<=nn;m++)
{
  l = npt;
  l = l / 2;
  while(mr+l > nn)
  l = l / 2;
  mr = (mr % l) + l;
  if(mr > m)
  {
  tr       = fr[m+1];
  fr[m+1]  = fr[mr+1];
  fr[mr+1] = tr;
  ti       = fi[mr+1];
  fi[m+1]  = fi[mr+1];
  fi[mr+1] = ti;
  }
}
l = 1;
while(l < npt)
{
  istep = 2*l;
  el = l;
  for(m=1;m<=l;m++)
  {
  a  = 3.1415926535 * (double)(1-m) / el;
  wr = cos(a);
  wi = sin(a);
  for(i=m;i<=npt;i=i+istep)
  {
    j = i + l;
    tr = wr * fr[j] - wi * fi[j];
    ti = wr * fi[j] + wi * fr[j];
    fr[j] = fr[i] - tr;
    fi[j] = fi[i] - ti;
    fr[i] = fr[i] + tr;
    fi[i] = fi[i] + ti;
  }
  }
  l = istep;
}

j = 1;

for(i=1;i<=npt;i=i+2)
{
  littlearray[i]   = fr[j]/(double)(npt/2);
  littlearray[i+1] = fi[j]/(double)(npt/2);
  j = j + 1;
}

if(npt>8995)
```

```
  {
    printf("\nERROR npt = %ld",npt);
    exit(0);
  }
  fftcount = 0;
  for(i=0;i<=npt-2;i=i+2)
  {
    fftval = (( (littlearray[i]*littlearray[i]) + (littlearray[i+1]*littlearray[i+1])) );
    temparray[fftcount] = fftval;
    fftcount++;
    if(fftcount >= 35999)
    {
    printf("\n fftcount = %ld",fftcount);
    exit(0);
    }
  }
  for(i=0;i<=fftcount;i++)
    littlearray[i] = temparray[i];

}
```

GenNet.bas

```
f1$="t12.fft"
f2$="ff12.fft"
f3$="chop1.dat"
f4$="chop2.dat"

dim f1(80)
dim f2(80)
cls
fmax=20!

ch1=1!
ch2=1!
f2=0!

open f1$ for input as #1
open f3$ for input as #2
open f2$ for output as #3
open f4$ for input as #4

max=0
rem read in and search for max first slice
for i = 1 to 80
   input #1, f1(i)
   if f1(i)>max then max = f1(i)
next i

rem normalize to max
for i = 1 to 80
   f1(i)=f1(i)/max
   print #3, using "#.######"; f1(i);
   print #3, " ";
next i

max=0
rem read in and search for max second slice
for i = 1 to 80
   input #2, f2(i)
   if f2(i)>max then max = f2(i)
```

```
next i

rem normalize to max
for i = 1 to 80
    f2(i)=f2(i)/max
    print #3, using "#.######"; f2(i);
    print #3, " ";
next i

REM INPUT TIME AND FREQUENCY FOR ONE
input #2, t, f
rem input time and frequency for two
if ch2 = 1! then input #4,t,f2
    print #3, using "#.######"; ch1;
    print #3, " ";
    print #3, using "#.######"; f;
    print #3, " ";
    print #3, using "#.######"; ch2;
    print #3, " ";
    print #3, using "#.######"; f2;
    print #3, " ";

for j = 1 to 279
    print j
    max =0
Rem find internormalization factor
    for i = 1 to 80
        f1(i)=f2(i)
        input #1,f2(i)
        if f2(i)>max then max = f2(i)
    next i
Rem output f1
    for i = 1 to 80
        print #3, using "#.######"; f1(i);
        print #3, " ";
    next i
Rem internormalization
    for i = 1 to 80
        f2(i)=f2(i)/max
        print #3, using "#.######"; f2(i);
        print #3, " ";
    next i

Rem input time and frequency for one
    input #2, t,f:f=(f-8)/fmax
rem input time and frequency for two
    if ch2=1! then input #4, t,f2:f2=(f2-8)/fmax
    print #3, using "#.######"; ch1;
    print #3, " ";
    print #3, using "#.######"; f;
    print #3, " ";
    print #3, using "#.######"; ch2;
    print #3, " ";
    print #3, using "#.######"; f2;
    print #3, " ";
next j
close 1
close 2
close 3
close 4
```

Appendix B
Simulated Sources

In order to eliminate noise and effects that were associated with signals measured at field sites, and to provide any given case desired for testing, a computer code that would simulate a moving source was programmed. The code, which was written in Visual Basic, allowed the user to control the initial position, velocity, frequency, and phase of up to three sources, in addition to the positions of up to three microphones. The output of the these three microphones was recorded into three files in an ASCII format. A sample input screen from this program is shown in Figure B1.

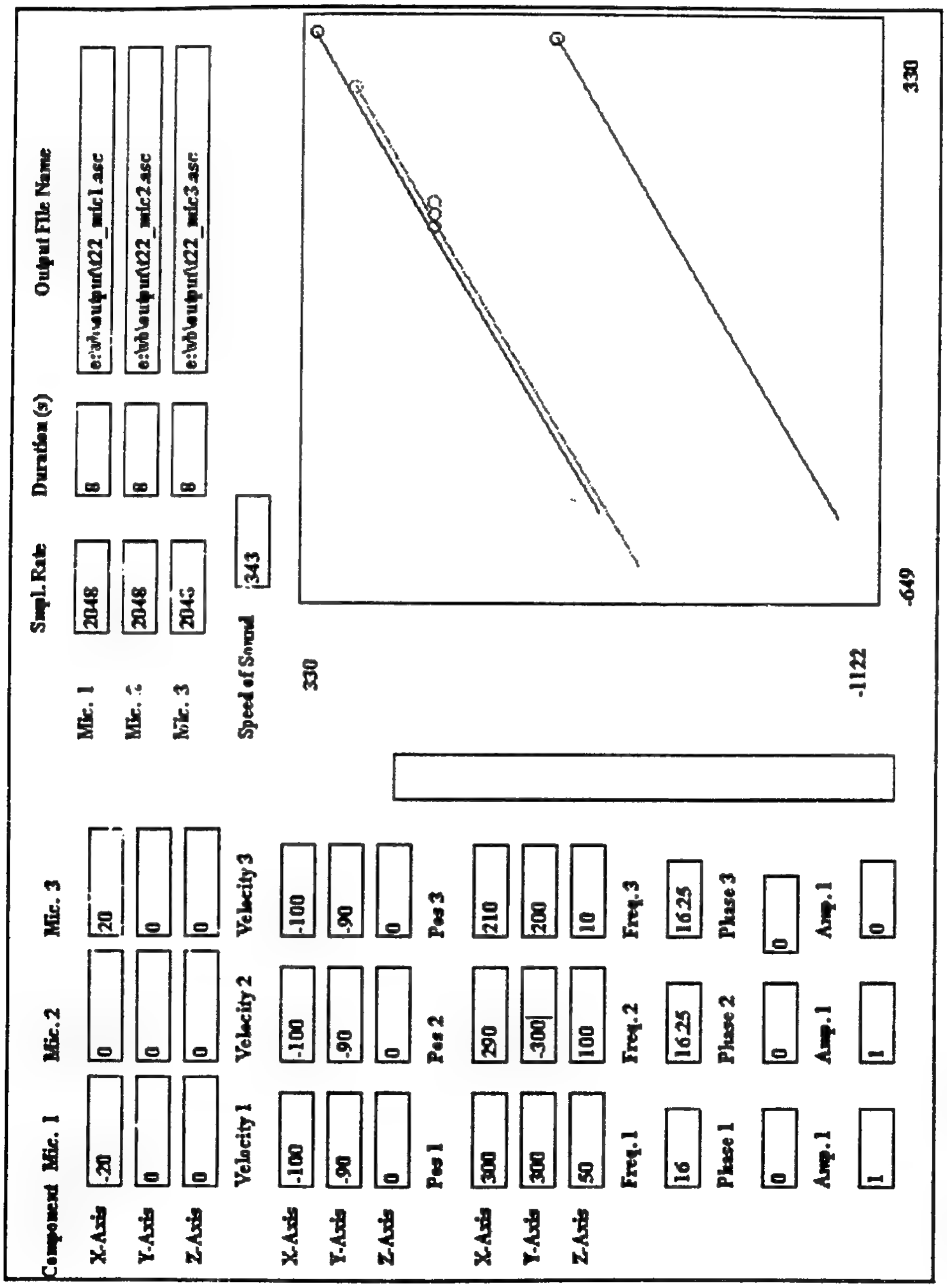


Figure B1. Sample of input screen from moving source simulation program

Appendix C
Data Used in ANN Analysis

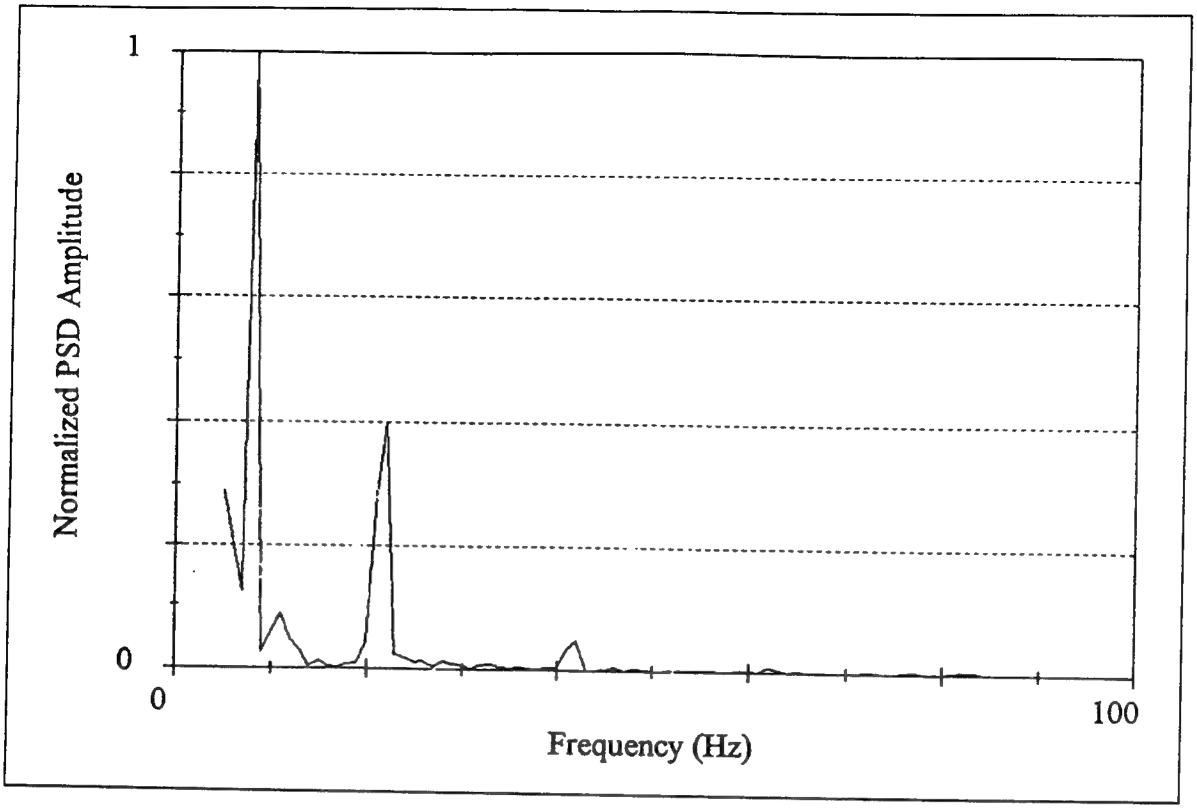


Figure C1. PSD from Case C used as sample number one

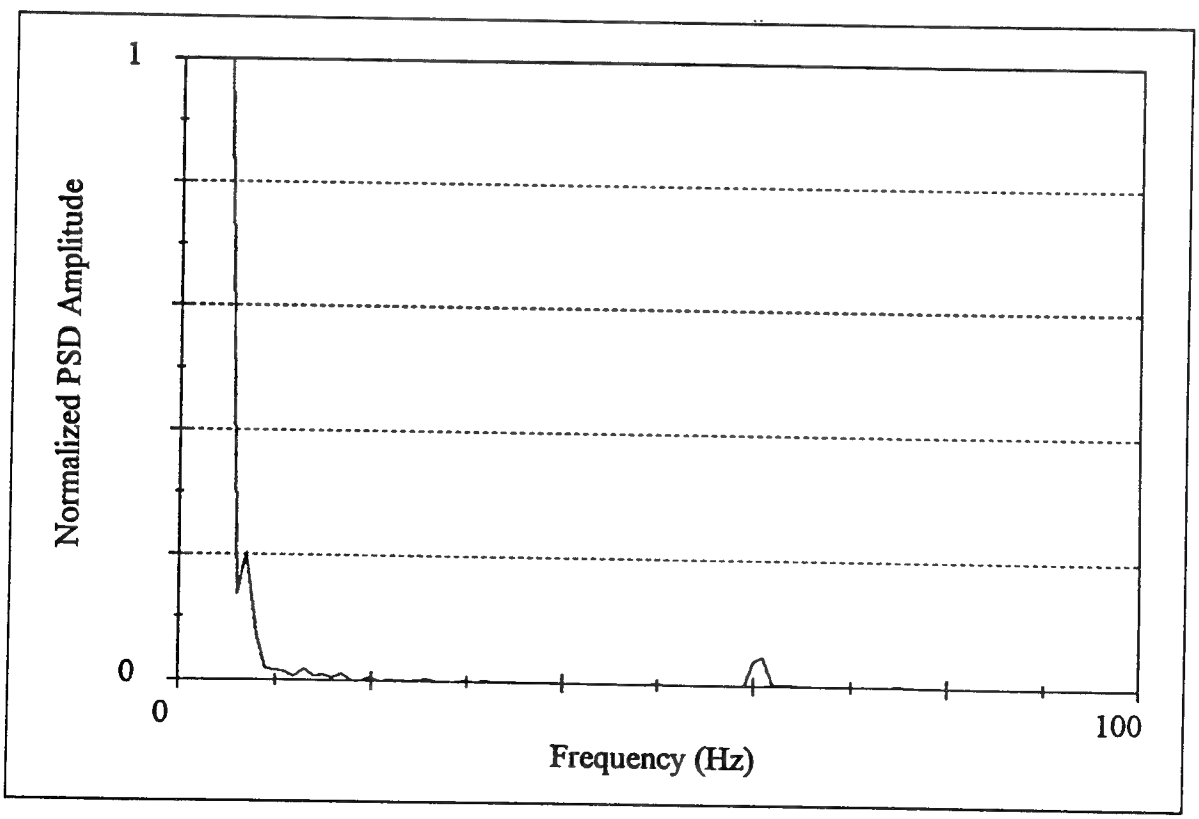


Figure C2. PSD from Case B used as sample number two

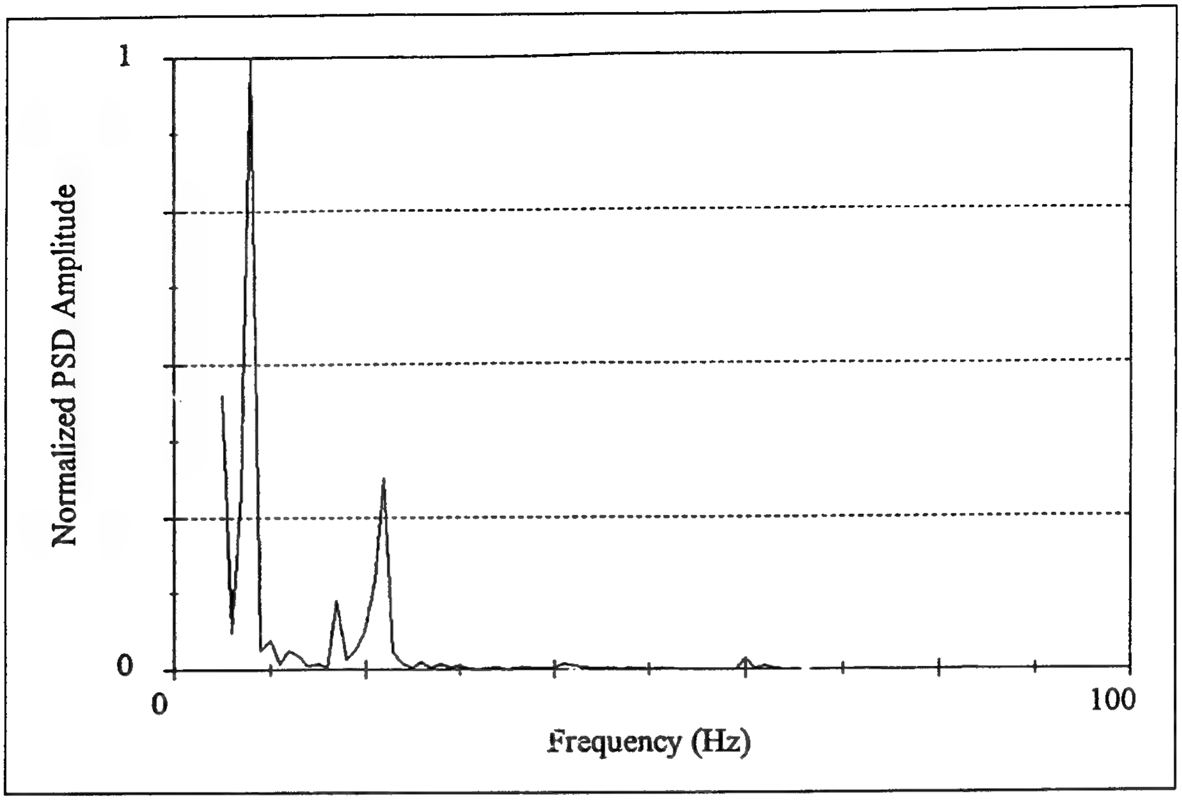


Figure C3. PSD from Case I used as sample number three

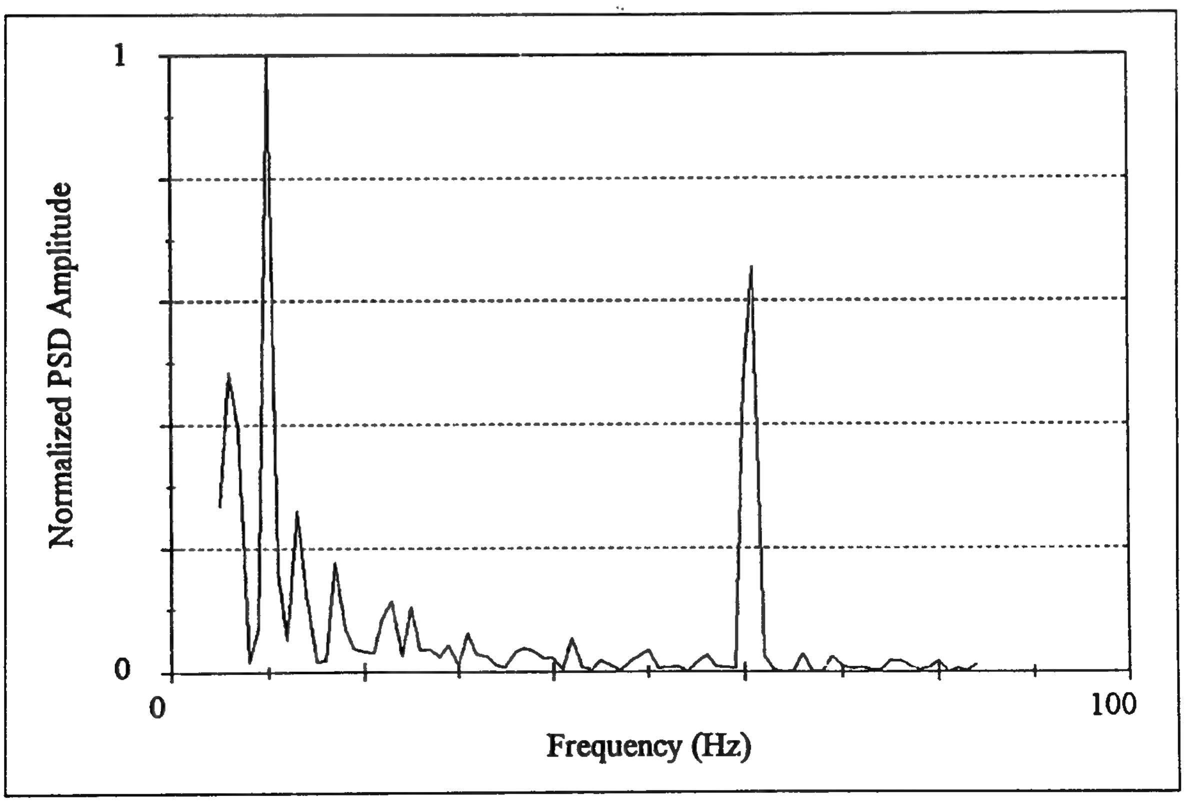


Figure C4. PSD from Case B used as sample number four

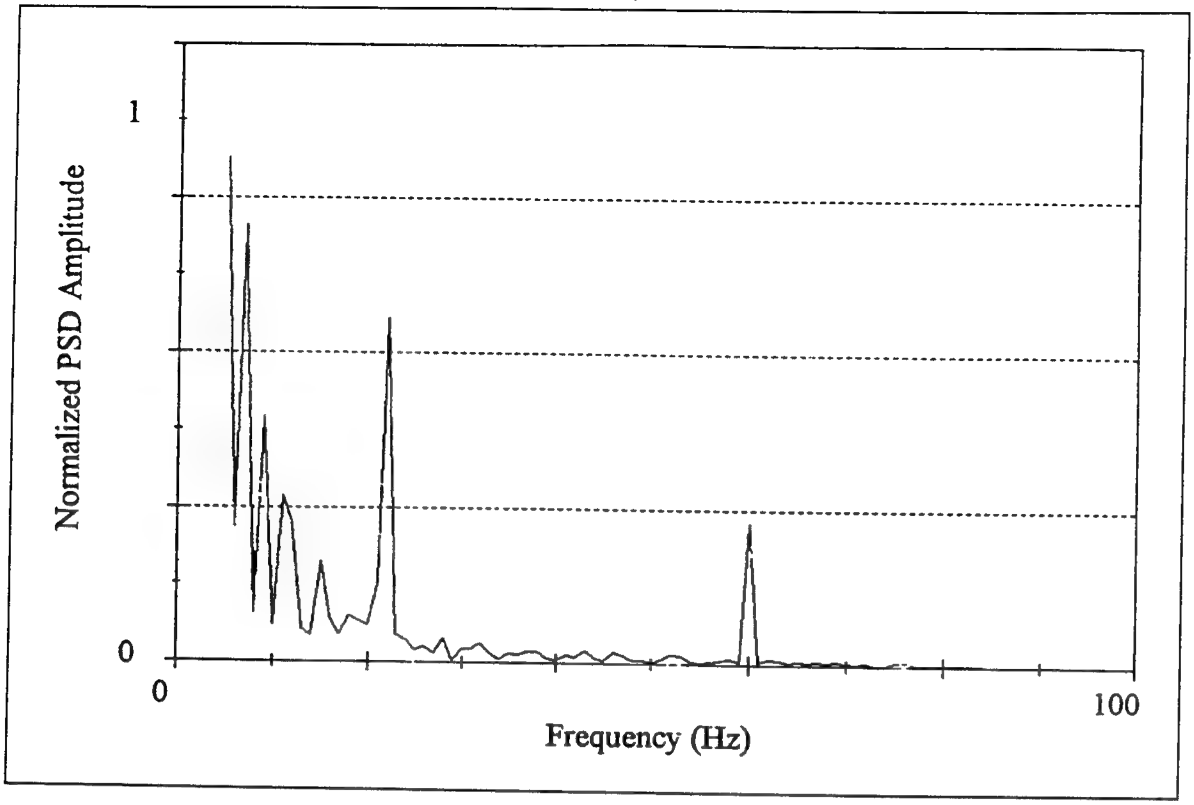


Figure C5. PSD from Case H used as sample number five

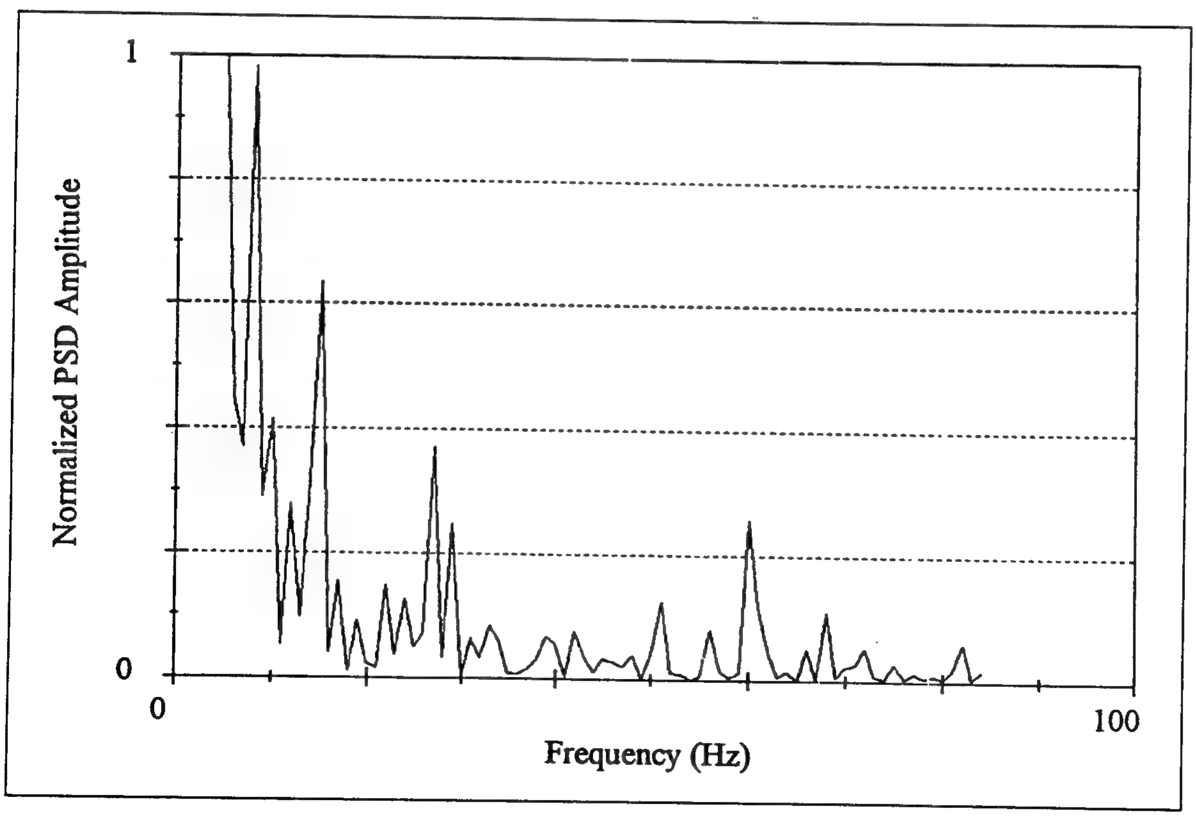


Figure C6. PSD from Case B used as sample number six

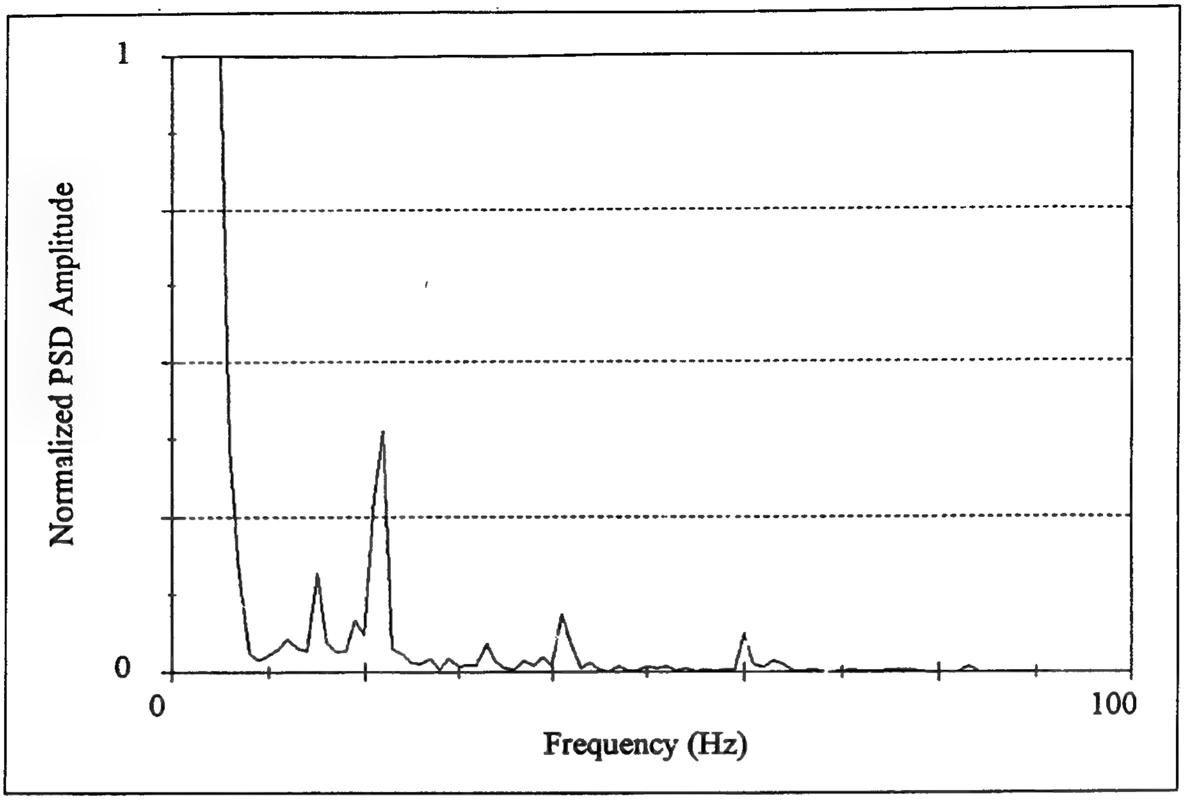


Figure C7. PSD from Case H used as sample number seven

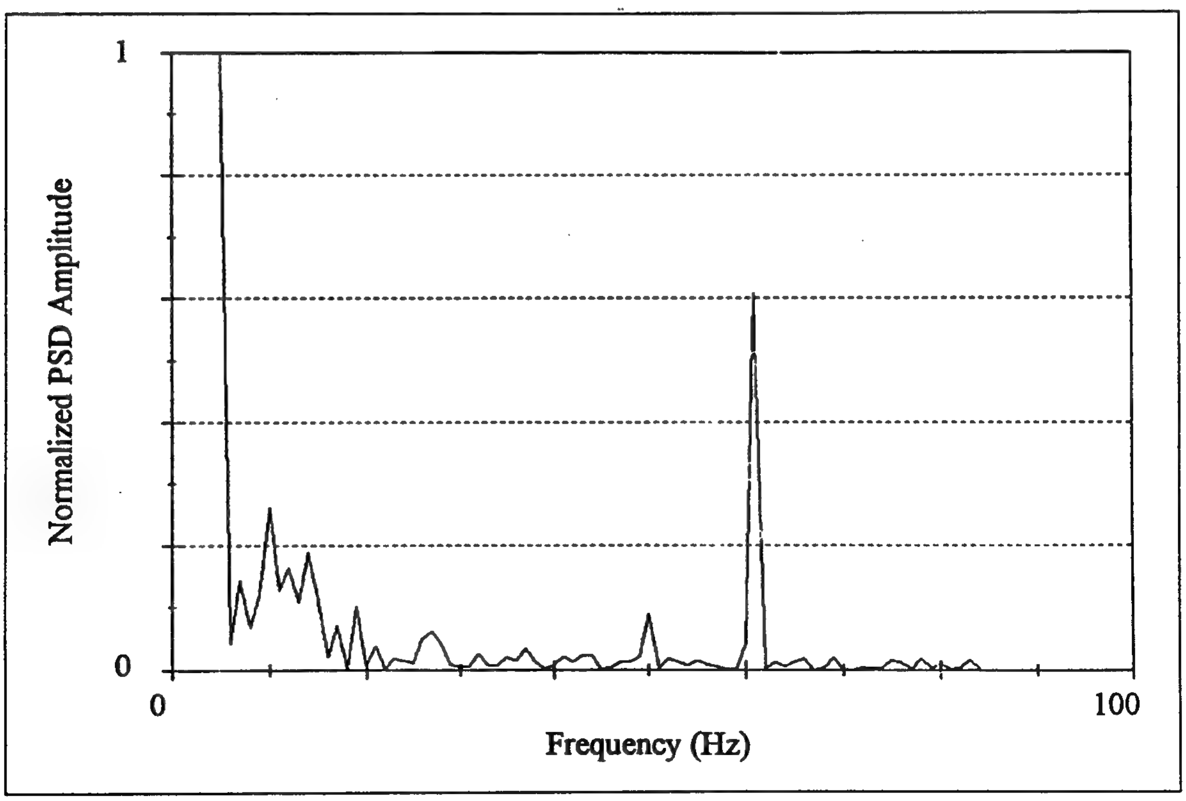


Figure C8. PSD from Case B used as sample number eight

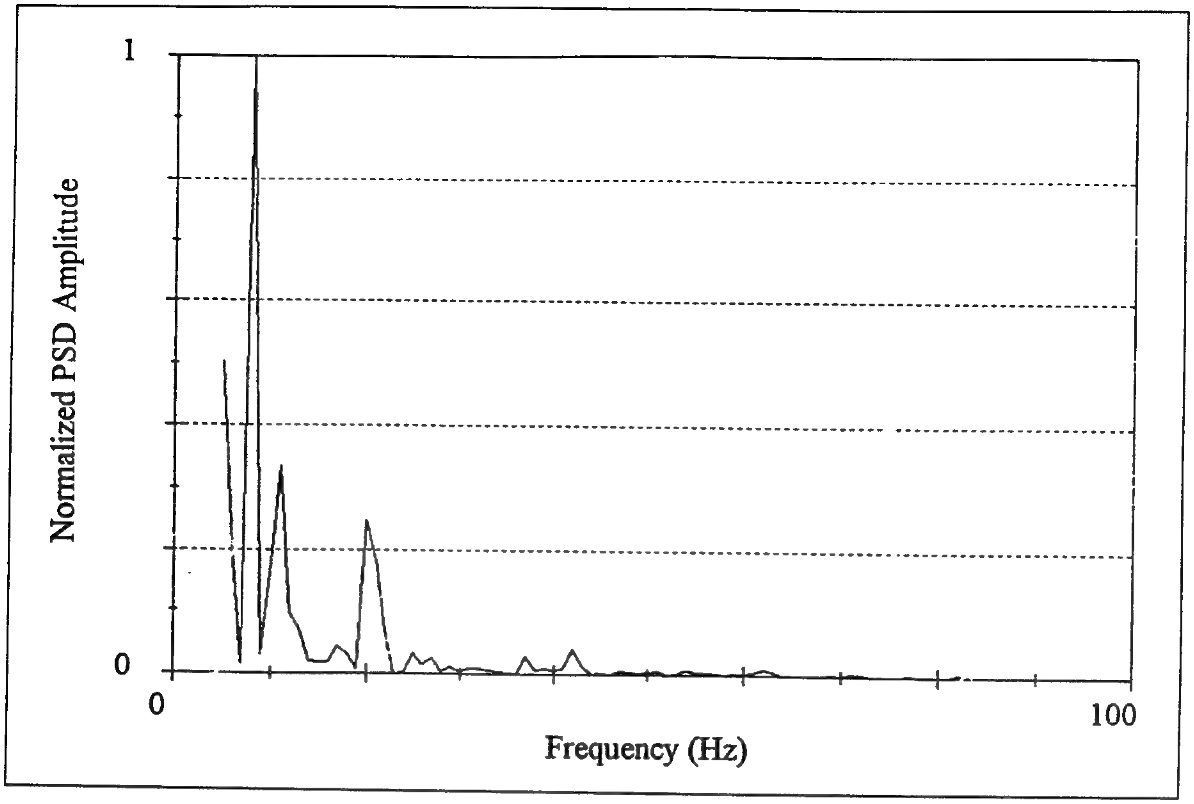


Figure C9. PSD from Case C used as sample number nine

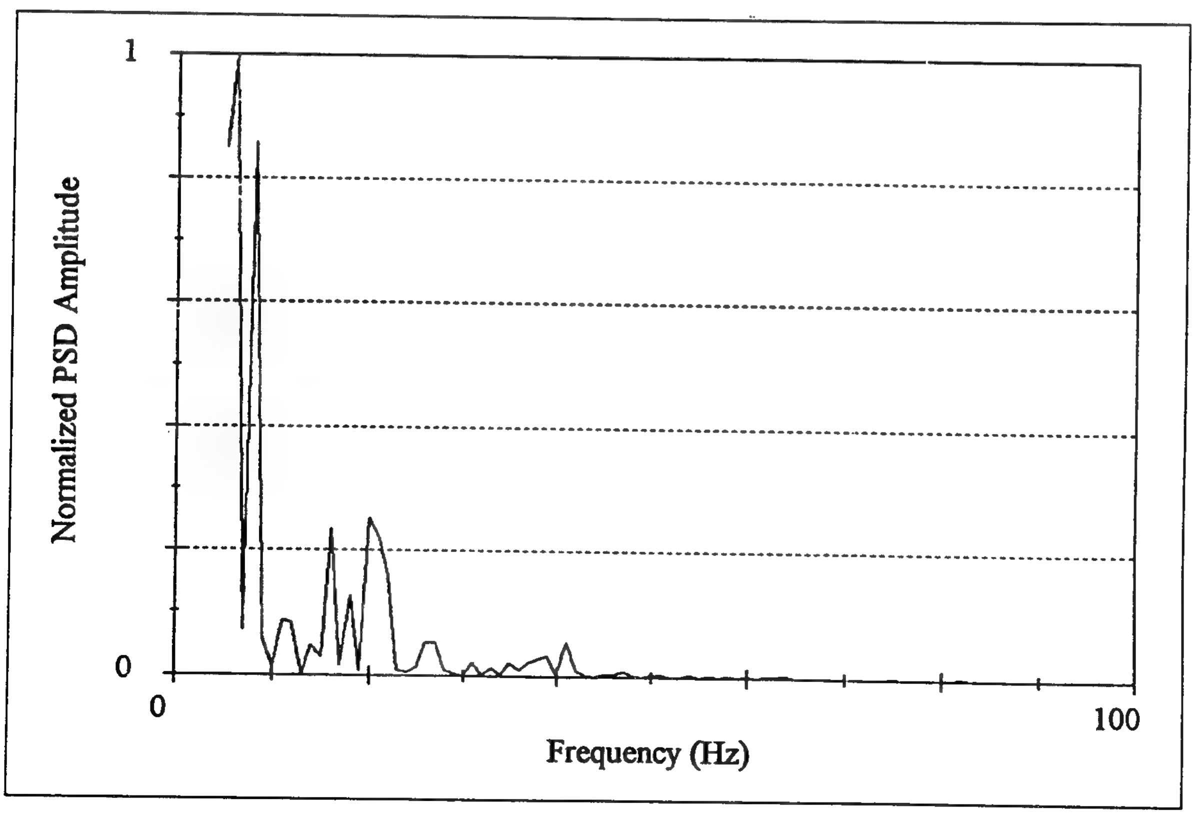


Figure C10. PSD from Case C used as sample number ten

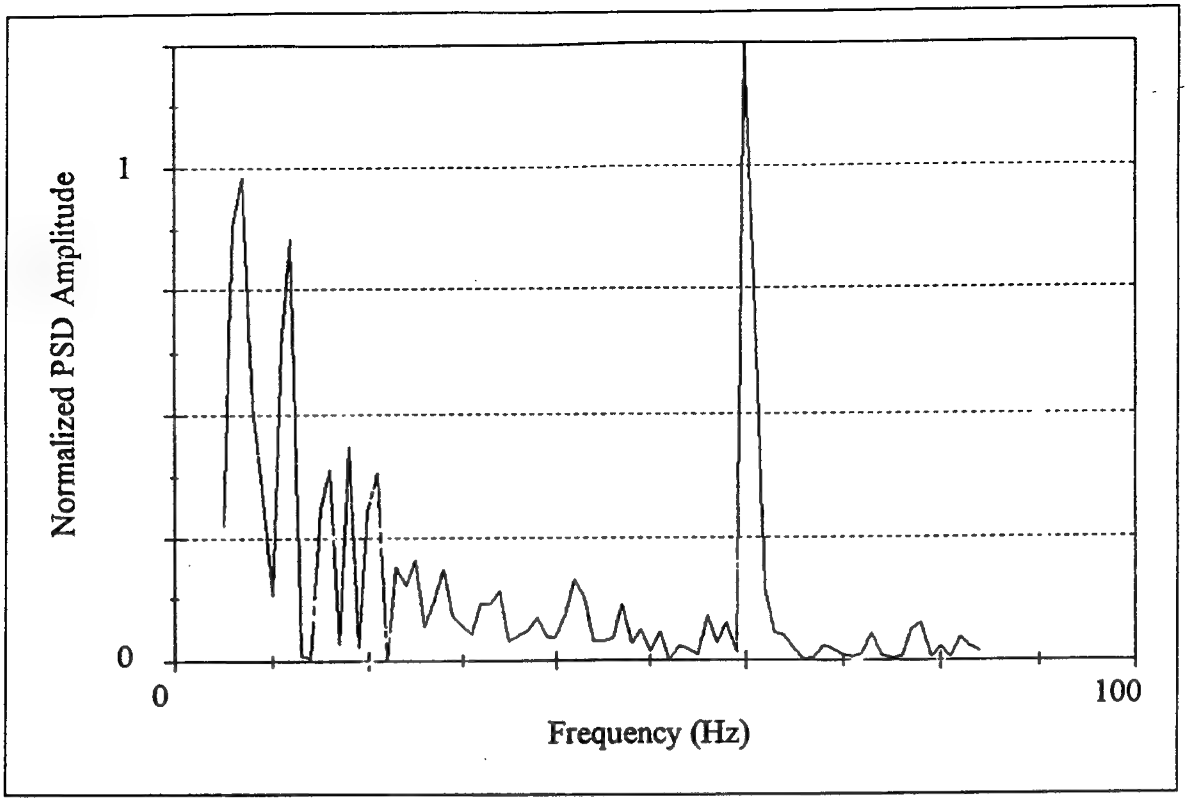


Figure C11. PSD from Case B used as sample number eleven

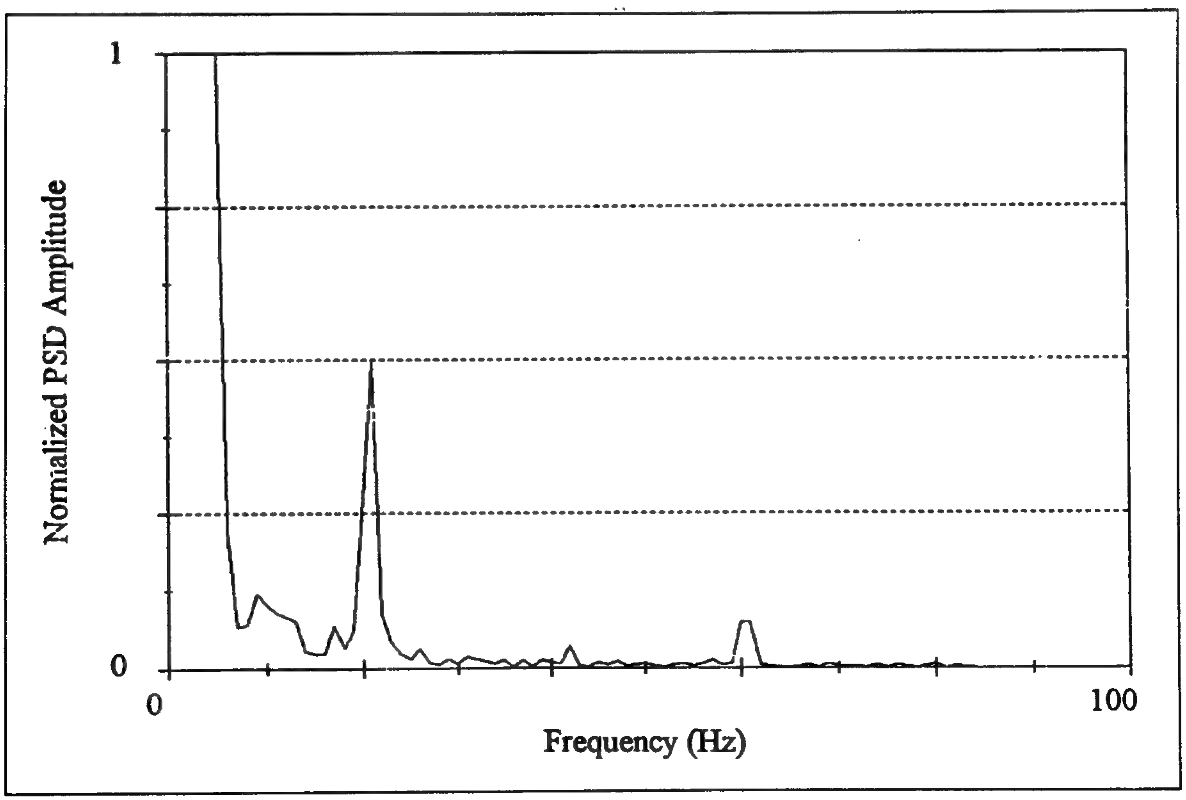


Figure C12. PSD from Case H used as sample number twelve

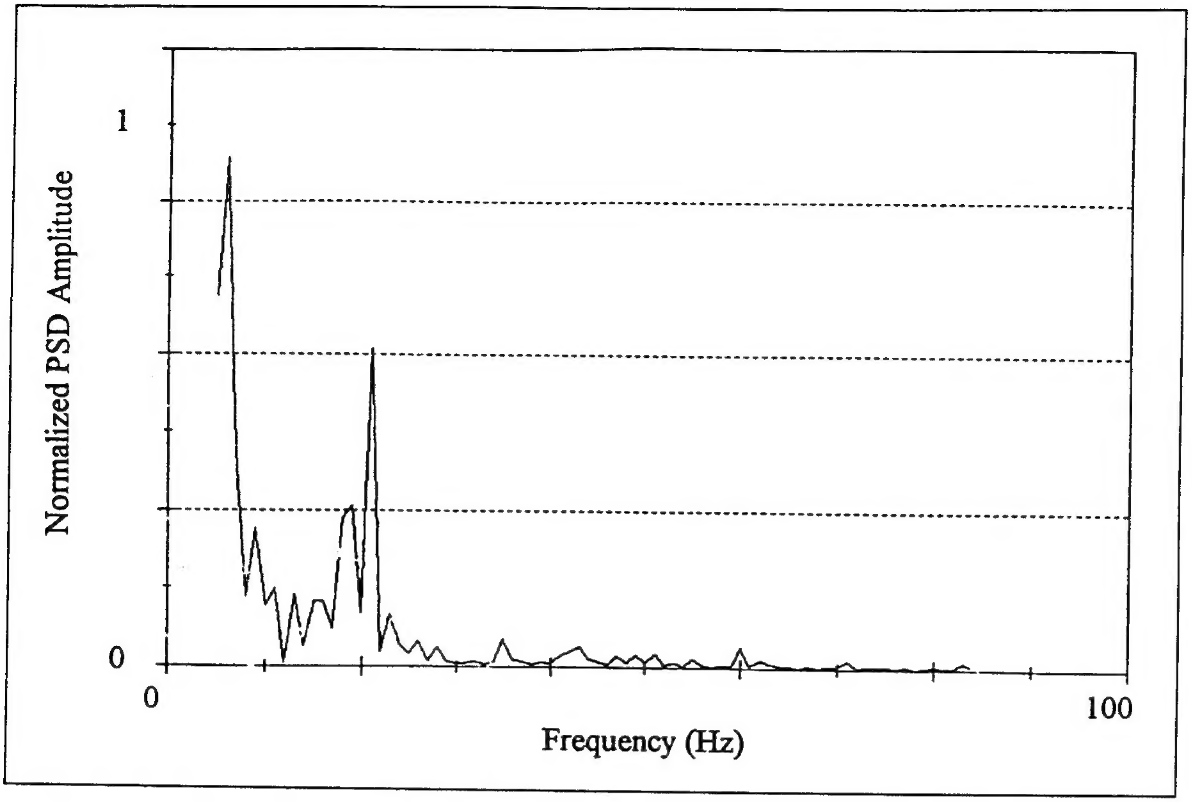


Figure C13. PSD from Case H used as sample number thirteen

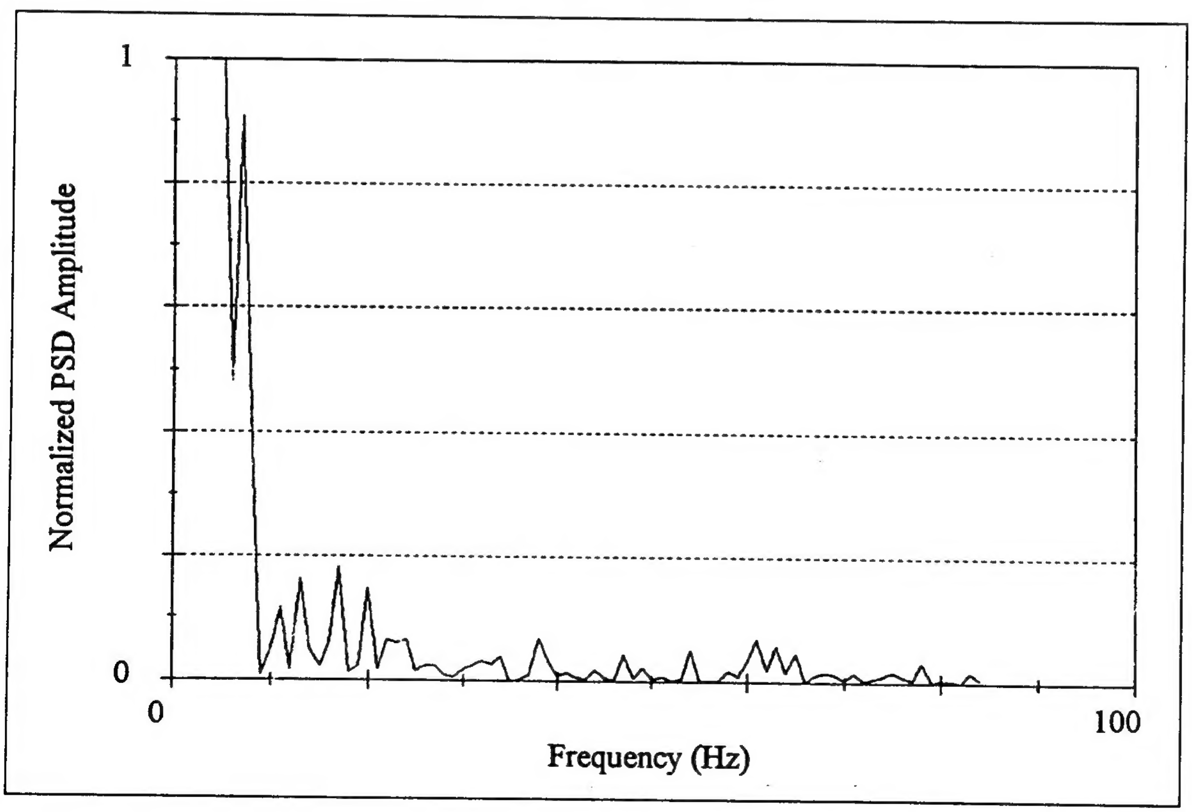


Figure C14. PSD from Case G used as sample number fourteen

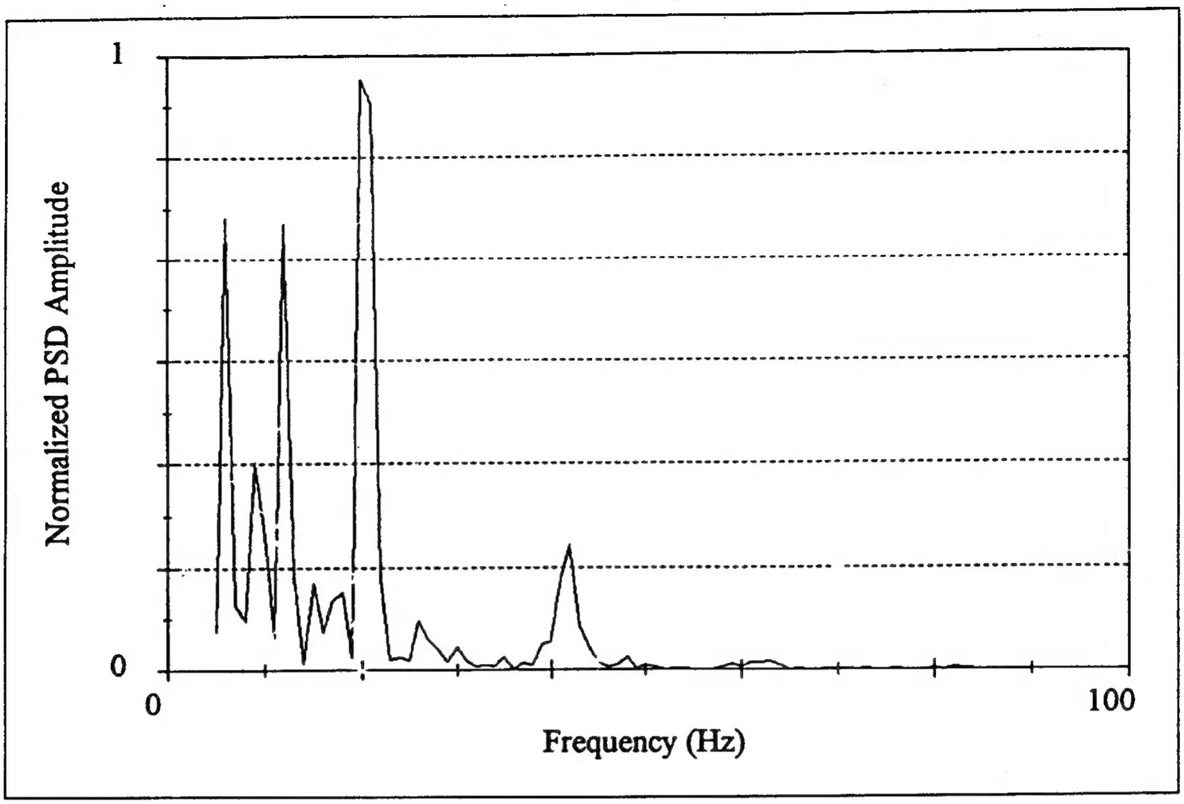


Figure C15. PSD from Case F used as sample number fifteen

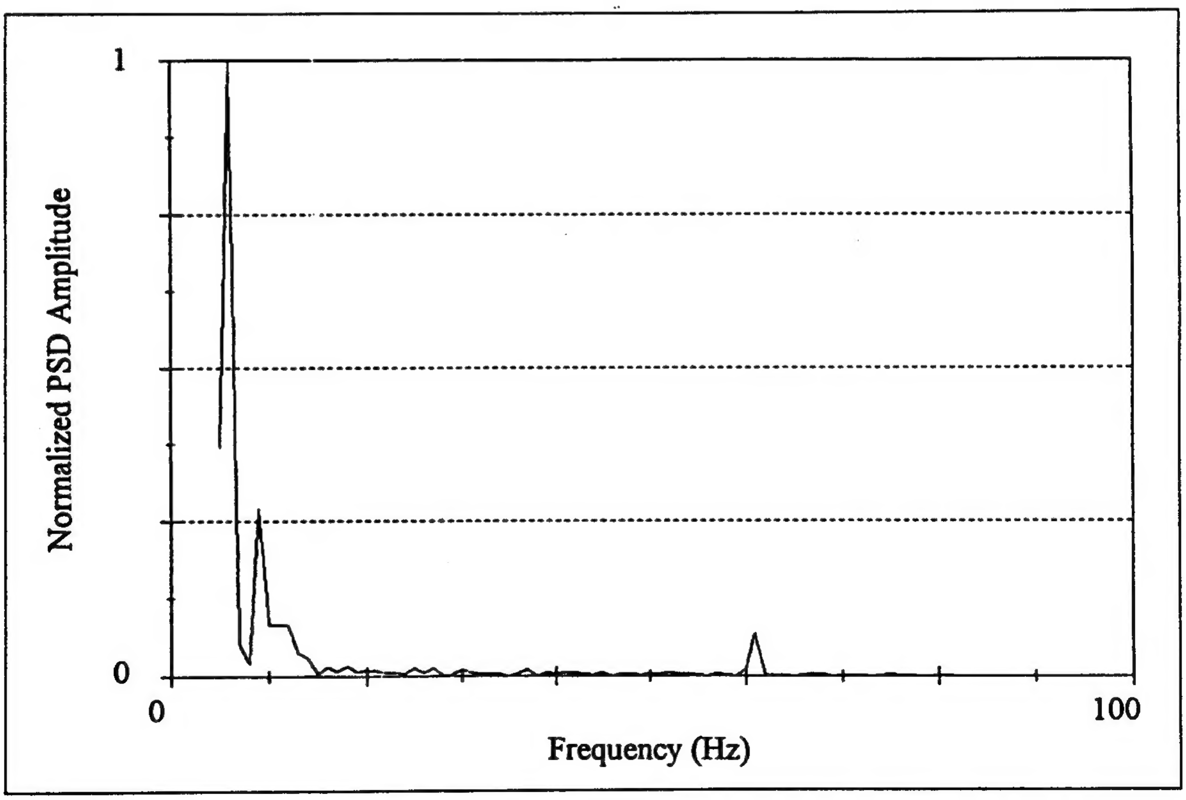


Figure C16. PSD from Case G used as sample number sixteen

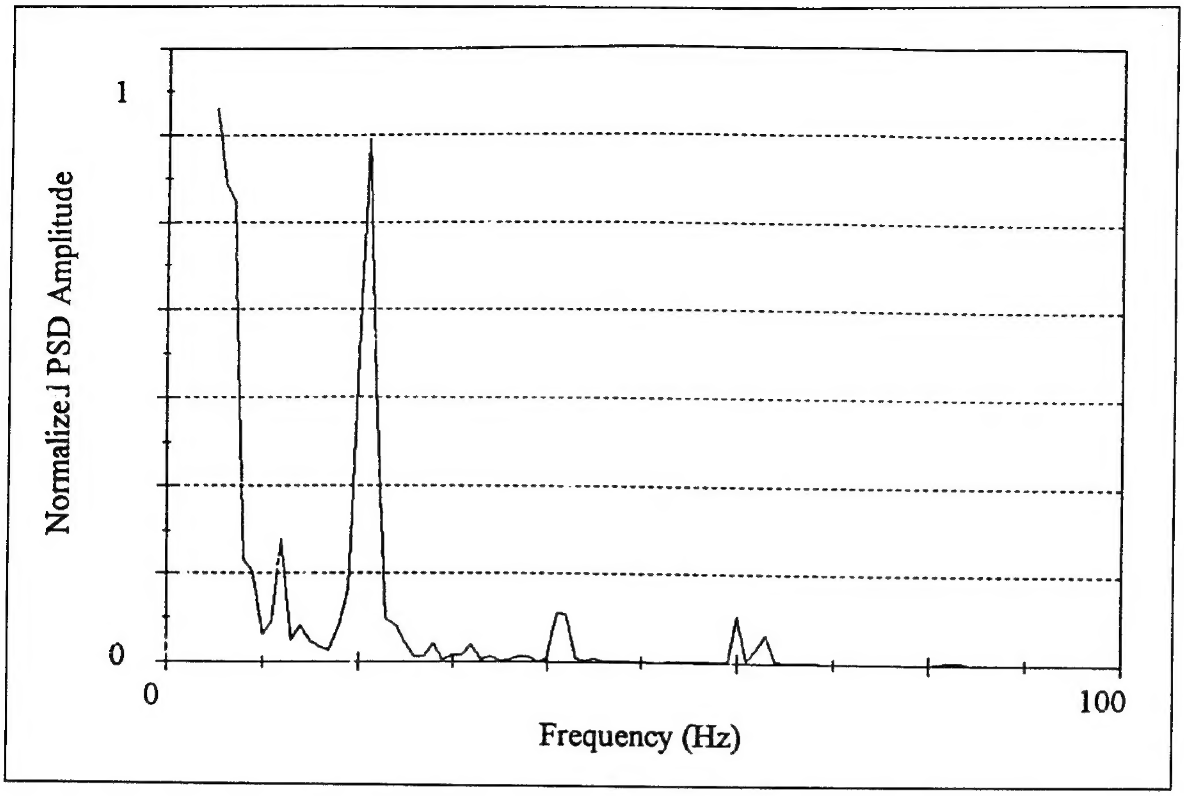


Figure C17. PSD from Case I used as sample number seventeen

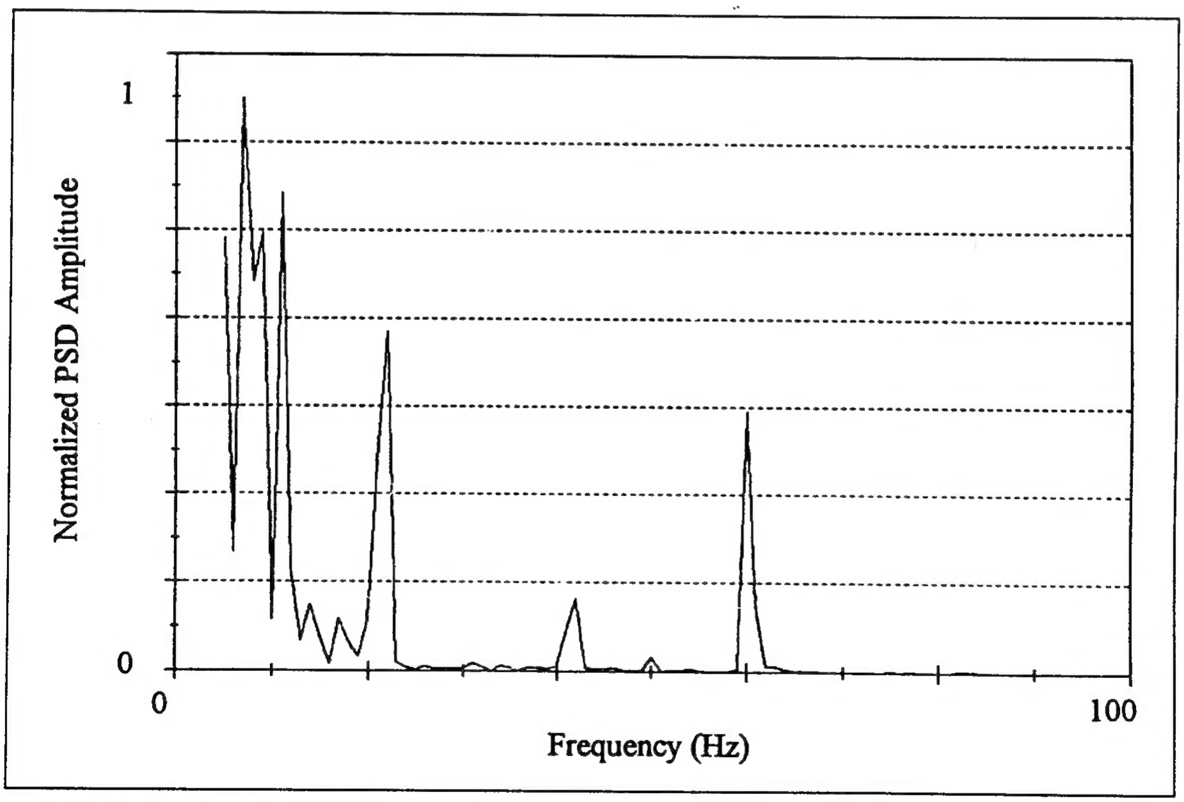


Figure C18. PSD from Case H used as sample number eighteen

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
March 1995

3. REPORT TYPE AND DATES COVERED
Final report

4. TITLE AND SUBTITLE
Evaluation of the Sensitivity and Limits of the Passive Acoustic Ranging Algorithm for Single and Multiple Helicopters

5. FUNDING NUMBERS

6. AUTHOR(S)
Benny L. Carnes and John C. Morgan

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Engineer Waterways Experiment Station,
3909 Halls Ferry Road, Vicksburg, MS 39180-6199
IIT Research Institute, 4140 Linden Avenue, Suite 201,
Dayton, OH 45432

8. PERFORMING ORGANIZATION REPORT NUMBER
Technical Report
SL-95-8

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Laboratory Discretionary Research Program
Assistant Secretary of the Army (R&D)
Washington, DC 20315

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Available from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Passive acoustic ranging (PAR), based on acoustic Doppler shifts, has been proposed as a suitable method for extracting velocity and range information from a source which emits a time-dependent acoustic signal. Operational constraints on the PAR method were evaluated and estimates were made of the limitations of realistic operation under various environmental conditions. In addition, performance of the PAR method was evaluated for multiple helicopters. Several methods of analysis of signals were tested to determine the optimum method for field implementation. It was found that additional information gained from preprocesssing with an artificial neural network would enable the analysis to process a signal from multiple moving sources and enhance the response time of the system.

14. SUBJECT TERMS
Acoustics
Helicopters
Passive detection and ranging

15. NUMBER OF PAGES
53

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED		